1651

TRS-80™ MODEL 100
A USER'S GUIDE

```
Mar 08,1983 Tue 10:37:42 (C)Microsoft
BASIC     TEXT      TELCOM    ADDRSS
SCHEDL    SCHEDL.DO NOTE.DO   CHAR.BA
-.-       -.-       -.-       -.-
-.-       -.-       -.-       -.-
-.-       -.-       -.-       -.-
-.-       -.-       -.-       -.-
Select: P;LPPPOOL_            -.-
           4874 Bytes free
```

Radio Shack
TRS-80
MODEL 100
PORTABLE
COMPUTER
F1 F2 F3 F4 F5 F6 F7 F8
PASTE LABEL PRINT BREAK PAUSE
ESC ! 1 @ 2 # 3 $ 4 % 5 ^ 6 & 7 7 * 8 8 (9 9) 0 — _ + = DEL BKSP
TAB Q W E R T Y U 4 I 5 O 6 P] [
CTRL A S D F G H J 1 K 2 L 3 : ; " ' ENTER
CAPS LOCK SHIFT Z X C V B N M 0 < , > . ? / SHIFT
GRPH CODE NUM

BY JOSEPH COLEMAN

TRS-80 MODEL 100
A USER'S GUIDE

BY JOSEPH COLEMAN

TAB BOOKS Inc.
BLUE RIDGE SUMMIT PA 17214

FIRST EDITION

FIRST PRINTING

Copyright © 1984 by TAB BOOKS Inc.

Printed in the United States of America

Reproduction or publication of the content in any manner, without express permission of the publisher, is prohibited. No liability is assumed with respect to the use of the information herein.

Library of Congress Cataloging in Publication Data

Coleman, Joseph.
TRS-80 model 100—a user's guide.

Includes index.
1. TRS-80 (Computer)—Programming. I. Title.
QA76.8.T18C67 1983 001.64 83-10568
ISBN 0-8306-0651-3
ISBN 0-8303-1651-9 (pbk.)

Contents

Introduction vi

PART 1 AN OVERVIEW 1

1 The Model 100 3
Start-Up—Features and Specifications—A Typical User's Day

2 Model 100 Users 7
Who Should NOT Buy the Model 100—Who SHOULD Buy the Model 100

3 Hardware 12
The Display—Keyboard—Memory—Microprocessor—Power Supply—Connections—Miscellaneous Buttons, Plugs, and Switches—Peripherals

4 Software 21
BASIC—TEXT—SCHEDL and ADDRSS—TELCOM

PART 2 TEXT 31

5 Typing with TEXT 33
RAM Files—Selecting Text—Typing a Document—Wraparound, Carriage Returns, and Scrolling—Key Typing Hints—Exiting the Document

6 Revising and Printing with TEXT 39
Label—Cursor Movement Commands—Revising—Find—Printing TEXT

7 **Managing TEXT Documents** 45
Copying the Document—Deleting the Document—Renaming the Document—Using TEXT with Cassette

8 **Hands-On TEXT** 50
Typing the Document—Copying the Document—Revising the Document—File Management—Printing the Document

PART 3 SCHEDL and ADDRSS 57

9 **Introduction** 59
Data Base Management—The Differences between SCHEDL and ADDRSS

10 **Creating Data Bases** 61
Planning a Data Base—Creating NOTE.DO—Creating ADRS.DO—Pointers on Typing ADRS.DO Records—Updating Data Bases—Taking Advantage of TEXT—A BASIC Reminder

11 **Using SCHEDL and ADDRSS** 66

PART 4 TELCOM 69

12 **Introduction** 71
A Crash Course in Communications—Connecting Computers—Some Other Terms You Should Know—Communications and the Model 100—Entry Mode—Terminal Mode

13 **Using the Modem** 81
Modem Communications—CompuServe—Logging On to CompuServe—Using CompuServe—Telenet and Tymnet

14 **Using the RS-232** 89
Connecting the Computers—Programming the Model III for Communications—Uploading from the Model 100—Model III Programs for Communicating with the Model 100

PART 5 BASIC 99

15 **Introduction** 101
Invoking BASIC—Running a BASIC Program from the Menu—Using the Function Keys in BASIC

16 **Housekeeping** 105
DATE$ and TIME$—Power and Start-Up Commands—Commands to Work with the Function Keys—RAM File Management—Memory Management

17 **Working with BASIC Programs** 114
Creating New BASIC Programs—Working with BASIC Programs—Using TEXT with BASIC

18 **Outstanding Features of BASIC** 121
Interrupt Commands—Sound—Display Control—The Open Command

19 Demonstration Programs 134
Good Housekeeping Program—Line Demo Program—A Sound Program—Alarm Clock Program

20 Enhancement Programs 140
Search and Replace Program—Address Writer

Appendix ASCII Code 145

Index 149

Introduction

This book is written for anyone—novice or expert—who owns or is thinking of buying Tandy Corporation's portable computer, the TRS-80 Model 100.

Many computer enthusiasts have called 1983 the year of the portable computer. On the heels of the famous Osborne portable, four companies—Epson, Gavilan, Sharp, and Tandy—fielded an entry in the portable sweepstakes.

It's quite possible that Tandy's Model 100 is the winner. Every review that I have seen has been a rave. It was Tandy who first made computers available to the public with the TRS-80 Model I. They seem to have done it again with the portable TRS-80 Model 100.

What's so good about the Model 100? In two words, integrated design. The hardware and software are hand in glove. It's a powerful computer in a small package that shows little sign of compromise.

At 11¾ by 8½ by 1⅞ inches, the Model 100 is about the size of a loose-leaf notebook and truly portable. It fits easily into an attaché with room to spare. The Model 100 compares in memory and processing capability to many desk-top computers. The liquid crystal display (LCD) displays 8 lines of copy, and each with up to 40 columns (characters). A contrast control makes it easy to see type on the display from any angle in almost any light. The typewriter-style QWERTY keyboard has eight programmable function keys, four command keys, and four cursor control keys and is better than the keyboards of some desk-top models. The microprocessor is the CMOS version of the 8085. It comes with 32K bytes of ROM and 8K to 32K bytes of RAM.

The Model 100 comes with five built-in, ready-to-run programs: TEXT, SCHEDL, ADDRSS, TELCOM, and BASIC, a powerful and uncompromising Microsoft BASIC.

Tandy seems to have accurately assessed the real software needs in a portable and designed the Model 100 to meet them. For example, it's doubtful that anyone would buy a portable computer to do state-of-the-art word processing, so the Model 100 provides very serviceable text editing but not the advanced features (such as reformatting and global search) that you would want in a desk-top model.

However, you would want a portable computer that was able to communicate with the home office or gain quick access to a data base. The Model 100 may be the best "smart" modem on the market.

This book on the Model 100 hardware and software will prove helpful in two main ways:

☐ If you are deciding whether to purchase a Model 100, the book enables you to assess it in terms of your needs.

☐ If you own a Model 100, the book supplements Tandy's user's manual (particularly on the topic of communications) and provides additional ideas and suggestions that will help you use the Model 100 to full advantage.

Here is a brief summary of what's in this book: Part 1 (Chapters 1 through 4) presents an overview of the Model 100. It describes its strengths and weaknesses as a portable computer, suggests some of the uses for which it is best suited, and takes you on a guided tour of the Model 100's hardware and software.

Parts 2 through 5 explain how to use the Model 100. They discuss each of the built-in programs and provide all the information you need to use them, plus some dandy ideas.

Chapters 19 and 20 in Section 5 provide exercises and suggestions you can use to test out the ideas in the book. For example, Chapter 20 contains a BASIC program you can write to perform a global search and replace on your text documents.

What you should not expect to learn from this book are the fundamentals of computing. If you are a novice, the Model 100 *is* a computer that you can use; however, in order to use the Model 100 to best advantage, you should be familiar with BASIC programming. This book will not teach you BASIC and assumes that you are at least acquainted with BASIC.

If you want to learn BASIC, I recommend Tandy's *Getting Started with TRS-80 BASIC*. If you know the rudiments of BASIC, this book will help you use your Model 100 to full advantage.

PART 1

AN OVERVIEW

The Model 100 is a powerful, well-designed portable computer. Chapters 1 through 4 provide you with an overview of the Model 100. After a brief description, Chapter 1 looks at an idealized day in the life of the Model 100 so you can get an idea of all the things it can do. Chapter 2 then discusses the different kinds of people who will probably find the Model 100 most useful, such as executives, traveling business persons and professionals, computer professionals and hobbyists, students, and writers. Chapters 3 and 4 take you on a guided tour of the Model 100's hardware and software.

Chapter 1

The Model 100

The Model 100 (Fig. 1-1) is a truly portable "notebook" computer. The same size as a loose-leaf binder, the Model 100 comes with 32K bytes of ROM (read only memory) that hold five built-in programs: BASIC (interpreter), TEXT, SCHEDL (Schedule), ADDRSS (Address) and TELCOM (telecommunications). In terms of user memory—RAM (random access memory)—you can buy from 8K bytes ($799) to 32K bytes ($1,134).

The display is a 40-character by 8-line liquid crystal display (LCD). The full-size, full ASCII typewriter-style QWERTY keyboard puts some desk-top models to shame. The Model 100 is powered by four AA batteries. With an optional ac adapter, you can plug it in to any outlet. When the batteries start to run down, a low-battery light comes on to tell you that you have fifteen minutes left before you will be out of power. If the batteries go dead, a rechargeable battery keeps your files (text and program) "alive" in memory for up to twenty hours.

START-UP

When you turn it on the Model 100, it is ready to go. The LCD displays the day, date, time, and amount of available memory. The Model 100 also displays a menu/directory (Fig. 1-2) lists the five built-in programs and any files saved in memory.

You can store two kinds of files in RAM: text documents and BASIC programs. For example, you store a text document and name it MEMO, the menu/directory automatically lists it as MEMO.DO (DO stands for "document"). If you name a BASIC program SORT and save it, the menu/directory lists it as SORT.BA (.BA stands for BASIC). To use a built-in program, run a BASIC program, or display a document, position the cursor on the document name and press the enter key.

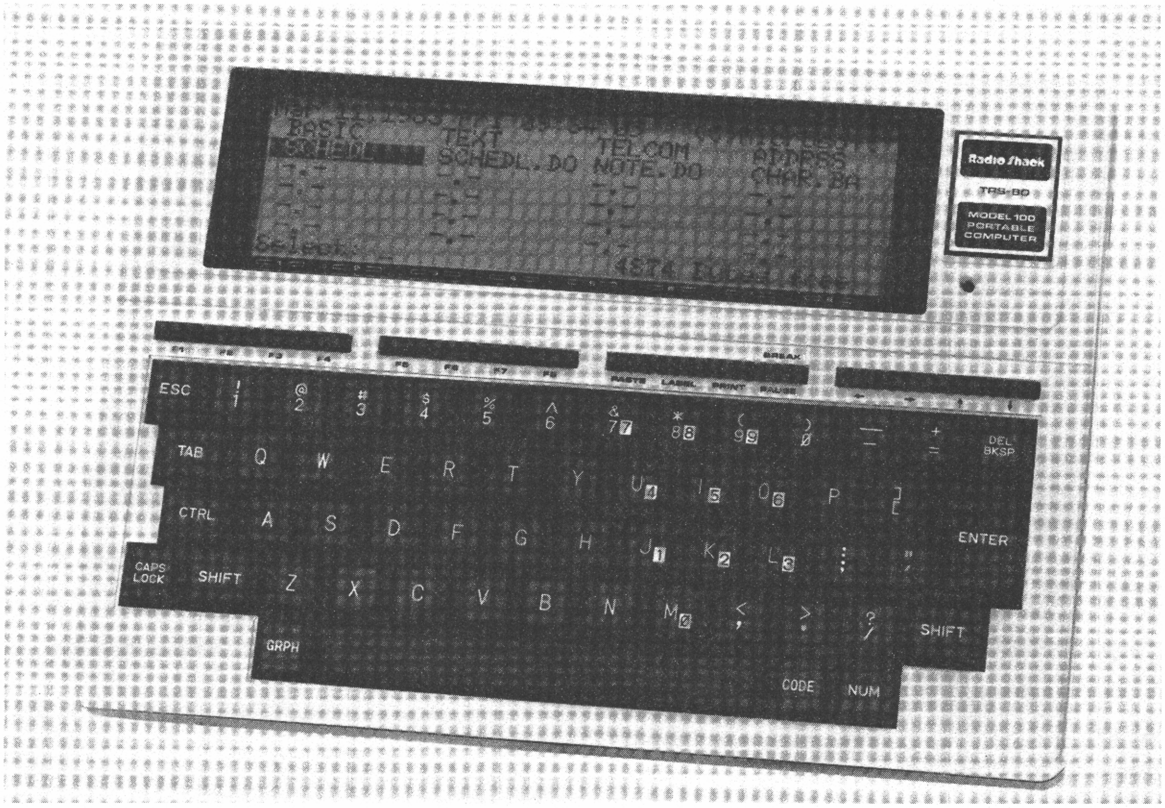
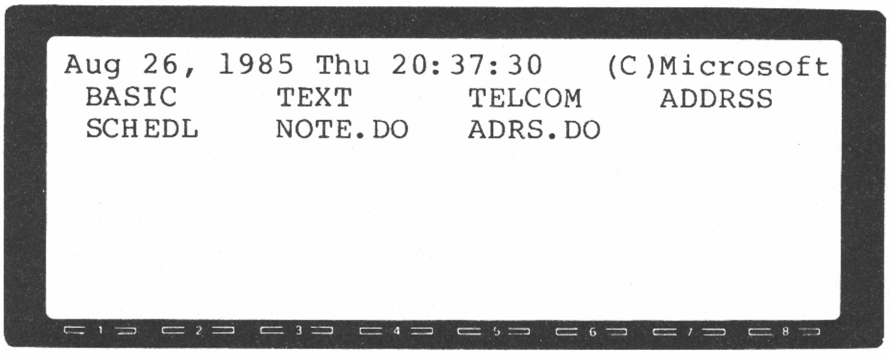


Fig. 1-1. The Model 100.



```
Aug 26, 1985 Thu 20:37:30   (C)Microsoft
 BASIC      TEXT       TELCOM    ADDRSS
 SCHEDL     NOTE.DO    ADRS.DO
```

Fig. 1-2. The Model 100's start-up display.

FEATURES AND SPECIFICATIONS

For convenient reference, major features and hardware specifications are listed below.

Uses. Personal and business computing, programming, and communications.

Size. 11¾ by 8½ by 2 inches.

Weight. 3½ pounds.

Display. Liquid crystal display (LCD); 8 lines by 40 columns (64 pixels by 240 pixels); graphics capabilities; white characters on black background or black characters on white background; adjustable screen contrast for easy viewing.

Keyboard. Full-size, full ASCII, typewriter-style QWERTY keyboard with 256 distinct characters and codes. In addition, there are eight programmable function keys, four command keys, and four cursor control (movement) keys.

Memory. 32K bytes of ROM; from 8K to 32K bytes of RAM. In addition Tandy plans to soon offer an additional 32K bytes of ROM containing application software.

CPU. A low-power CMOS version of the 8085, an 8-bit microprocessor operating at 2.4 MHz.

Interface Ports. Cassette, modem, RS-232-C, parallel printer, and bar-code reader. Figures 1-3 and 1-4 illustrate two of the ways the Model 100 can be interfaced to other equipment.

Power Supply. Four AA batteries or an optional ac adapter. A rechargeable nicad (nickel-cadmium) battery powers RAM when the power is switched off.

Software. A menu program plus five built-in application programs: BASIC, TEXT, ADDRSS (address), SCHEDL (schedule), and TELCOM (telecommunications).

Documentation. A 224-page, spiral-bound reference manual that provides instruction for general operations as well as on each of the five built-in programs. A 40-page, pocket-size *Quick Reference Guide*.

Other Features. Sound generator, real time clock.

Options. Hard carrying case; ac adapter; RS-232-C cable, parallel printer cable, cassette cable, telephone connector, acoustic coupler, and bar-code wand cable.

Price. Depending on the amount of RAM obtained, the Model 100 is priced at $799 for 8K RAM, $999 for 24K RAM, and $1,134 for 32K RAM.

Warranty. The usual Radio Shack 90 days against manufacturing defects.

A TYPICAL USER'S DAY

To give you an idea of the incredible range that Tandy has packed into this small package, let's look at a day in the life of computer expert Chip Hertz.

Chip Hertz owns a small computer-consulting firm. He must call on a client, Video Madness, in another city. Before Chip leaves the office, he uses TELCOM, the Model 100's telecommunications program, to load the client's records from his desk-top computer to the Model 100. He could also use a home-grown BASIC program.

On the train, Chip gets ready for tomorrow's meeting by reading the client's records on the Model 100's display. Next, Chip uses the TEXT program to type a few memos and letters. Finally, Chip runs a BASIC program to sort his list of outstanding invoices by due date. While the Model 100 is sorting the file, Chip decides to take a snooze. The same program that sorts his file will also sound an alarm at 5:30 so Chip won't miss his stop.

That night at the hotel, Chip hooks up the Model 100 to the telephone. The TELCOM program uses the ADDRSS program to automatically call Chip's computer back at the office and log on. To save on long distance charges, the Model 100 automatically transmits (uploads) Chip's memos and letters from the Model 100 to the office computer and then automatically receives (downloads) the messages from the office computer to its own memory. When the transmission is complete, the Model 100 logs off and hangs up the phone. At his leisure, Chip displays and reads the messages that the Model 100 received over the phone.

Before bed, Chip uses the Model 100's SCHEDL program to review his appointments for the next day; then he runs a BASIC program to wake him up in the morning and display the message, "don't forget to call Carla!"

At Chip's meeting with the client, he uses the TEXT program to take notes. It certainly doesn't hurt Chip's image as a computer consultant when he uses a portable computer for his notes!

The client wants to know how much a mainframe with a hard disk drive will cost. During lunch, Chip connects the Model 100 to the client's phone and using a local number logs on to a data base that lists computer prices. He downloads the data to the Model 100's memory. After lunch he is all ready to discuss pricing with his client.

After the meeting, Chip types a letter of agreement on the Model 100 and sends it back to the office. A hard copy of the draft will be waiting for him when he returns.

Back on the train after a successful meeting, Chip relaxes by playing backgammon with the Model 100. (The Model 100 is an excellent player, but since it is terrible at collecting its winnings, Chip plays for $1,000 a point.)

That should give you a general and idealized idea of how a computer professional might use the Model 100, but Chapter 2 shows how other people can benefit from owning one.

Chapter 2

Model 100 Users

The Model 100 is a great portable computer. If you want or need a portable, don't delay. Rush out and buy one today. However, if your idea of a portable includes portable printing and disk drives, then the Model 100 is not for you. If you are new to computing, the Model 100 *may* not be right for you.

WHO SHOULD NOT BUY THE MODEL 100

Two groups of people should not buy the Model 100. They are those with limited money to spend who are buying their first computer and those who consider a portable printer or disk drive a necessity.

The Model 100 as a First Computer

If you are buying your first computer and have limited resources, you probably should not buy the Model 100. True, the Model 100 is a great computer, and dollar for dollar it surpasses many desk-top models. True, no computer is easier to start up than the Model 100. True, you can take the Model 100 anywhere. For someone learning to compute, however, the small display can be a problem. For a *portable,* 8 lines by 40 characters is large, but I think it would be difficult to learn BASIC when you can view only 8 lines of your program at a time.

On the other hand, if you want to limit your initial investment to $799 and want a good introduction to what a computer can do, the Model 100 will provide you with more computer power than some models at twice the price. If communications is a high priority, the Model 100 is easier to learn and use than most desk-tops. However, you will probably want to buy a desk-top later.

The Printerless and Diskless Model 100

Some portable computers come with printers and disk drives. A few portables, such as the Epson, come with built-in printers. Some, such as the Osborne, come with built-in disk drives. The tradeoffs are obvious. The more mechanical devices you add to the portable, the heavier (read "less portable") it is.

The printer-equipped Epson weighs only four pounds, but the printer is tiny. Serious printers add serious weight and challenge the concept of portability.

Disk drives come in only one size—heavy. The famous Osborne portable, for example, is equipped with two disk drives. It weighs 28 pounds. It's portable the way a small color television is portable.

Although you can easily hookup your Model 100 to a cassette recorder, you may choose not to. Because the Model 100's memory is continuously powered, the computer will retain text or programs until you are within cable or telephone reach of your desk-top computer and can then print them or store them on disk.

However, if you feel that a portable computer needs a portable printer or disk drive, then the Model 100 is not your machine. There are, however, rumors that Tandy is coming out with portable disk drives to go with the Model 100.

WHO SHOULD BUY THE MODEL 100

If you want a truly portable computer, buy the Model 100. It is at its best a companion to a desktop computer at home or at the office. The Model 100 is ideal for the executive who, whether or not

Fig. 2-1. The Model 100 is an excellent computer for the executive.

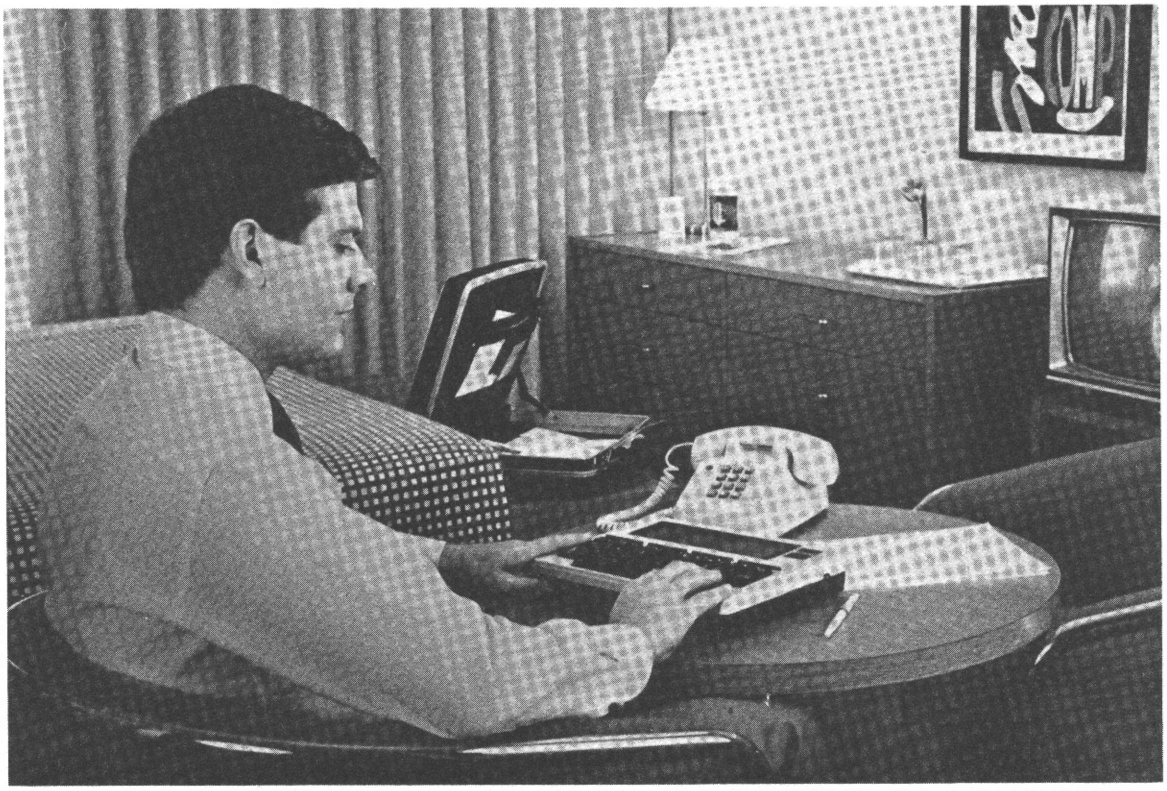


Fig. 2-2. The Model 100 is small enough to fit in a briefcase, portable for carrying anywhere.

he has a terminal at his desk, wants a powerful work station at his fingertips.

The Model 100 is ideal for these people: executives, traveling business persons and professionals, computer professionals and hobbyists, writers and journalists, and students.

Executives

The executive will find SCHEDL and ADDRSS, the Model 100's schedule and address programs, extremely helpful for managing day-to-day commitments and for keeping track of names, addresses, and expenses. With the Model 100's telecommunications program, TELCOM, you can gain instant access to a wide range of information from stock quotes to pork belly futures. BASIC programs enable the executive to use the Model 100 as a decision-making tool, displaying an endless variety of charts, graphs, projections, cash flows, quotes—you name it. The text program, TEXT, enables the executive (Fig. 2-1) to take notes and write letters or memos.

Travelers

For the traveling business person or professional (Figs. 2-2 and 2-3) with a computer back at the office, the Model 100 will, no doubt, provide an unsurpassed portable work station. On the road, the Model 100 provides a record of expenses and meetings, serves as a note pad, a reference tool, and enables the user to communicate over the phone with the home office.

Computer Professionals

If you are a computer professional or hobbyist, buy a Model 100. You and your computer will never have to suffer "separation anxiety" again. Take the Model 100 with you everywhere, work on your programs, then upload your files to your main system when you get home.

Students

If you are a student, buy a Model 100. If your typing is better than your writing, you can use the Model 100 in class to take notes. At home you can edit your notes and print them out. Use it to write term papers. Use your Model 100 at the library to take notes. Then, when you write the paper, just

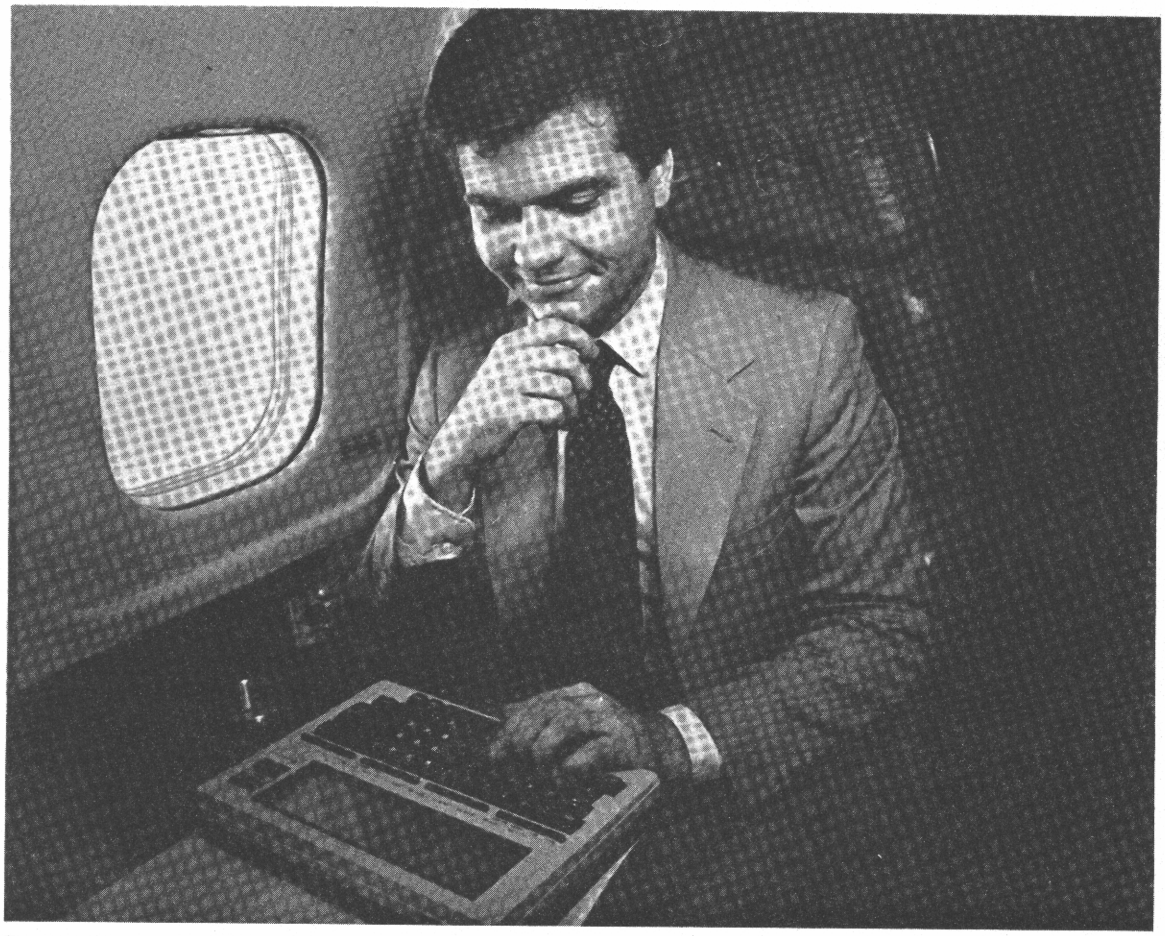


Fig. 2-3. The Model 100 is small enough to use comfortably on an airplane.

"cut" and "paste" your notes directly into your paper using the computer's unique editing features.

Writers and Journalists

If you are a writer with a word processor, or a journalist who uses a terminal, buy a Model 100. You can take the Model 100 anywhere, e.g. the library or to an interview. If inspiration strikes, pull out the Model 100 and start writing. When you get back to your word processor or terminal, upload your files. (Warning: the Model 100 may rob you of some excuses for procastinating.)

Chapter 3

Hardware

The hardware you get when you buy a Model 100 is quite portable. While researching this book, I flew from New York to North Carolina for my sister's wedding. In my attaché case I put the Model 100, the user's manual (about the same size as the Model 100), a notebook, some papers, and two paperback books. I had room to spare. (I must confess that I enjoyed the curious stares of the other passengers as I computed at 33,000 feet.)

The Model 100 is 11¾ by 8½ by 1⅞ inches, and it weighs only 3½ pounds.

THE DISPLAY

The display is less than optimum size for high-powered word processing, but who buys a portable for high-powered word processing? The liquid crystal display (LCD) can display either black characters on a white background or a reverse field, white characters on a black background called reverse video.

From my experience with digital watches and electric typewriters, I had always looked down my nose at LCDs. I had assumed that they were limited to a few characters, and that characters were hard to see at certain angles and in some kinds of light. The Model 100 has changed my attitude on both counts. As of this writing, the Model 100 has the largest LCD of any portable computer. The window is 7⅜ inches by 2⅛ inches and displays eight lines of forty columns (more than some low-budget home computers). If you are interested in graphics, you can use the LCD as a dot matrix measuring 240 pixels by 64 pixels.

The Model 100 easily solves light source and viewing angle problems (common to some LCDs) with a DISP control wheel (Fig. 3-2) that adjusts contrast.

The display is more versatile than those found on many desk-top models. It provides the full set of uppercase and lowercase characters, foreign-

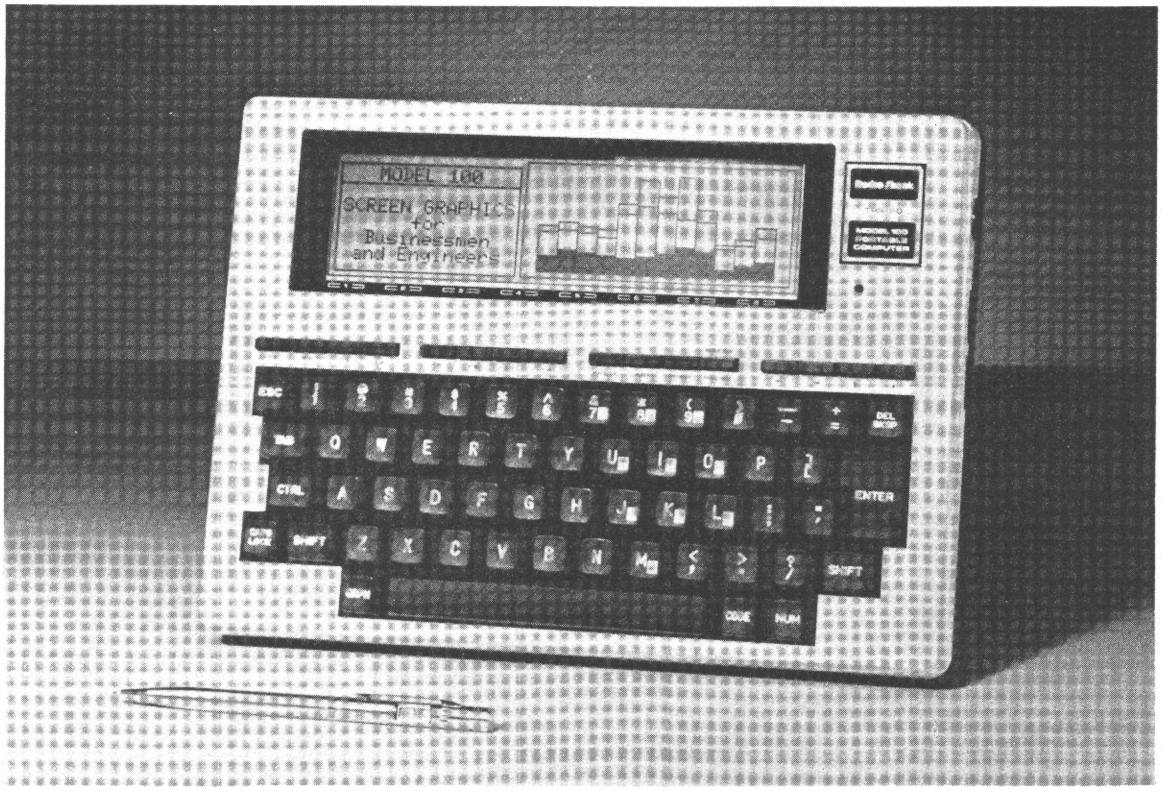


Fig. 3-1. The Model 100's size makes it truly portable.

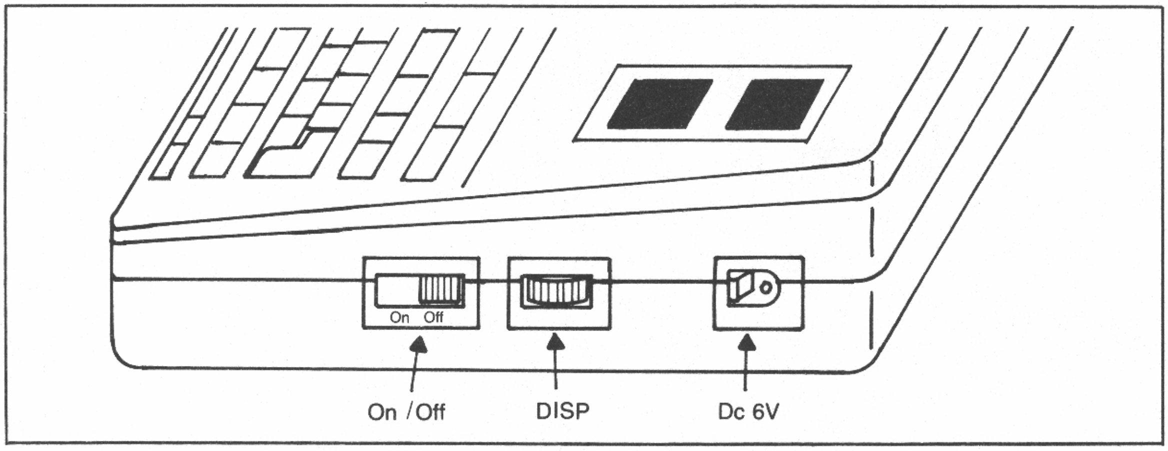


Fig. 3-2. Right side view of the Model 100.

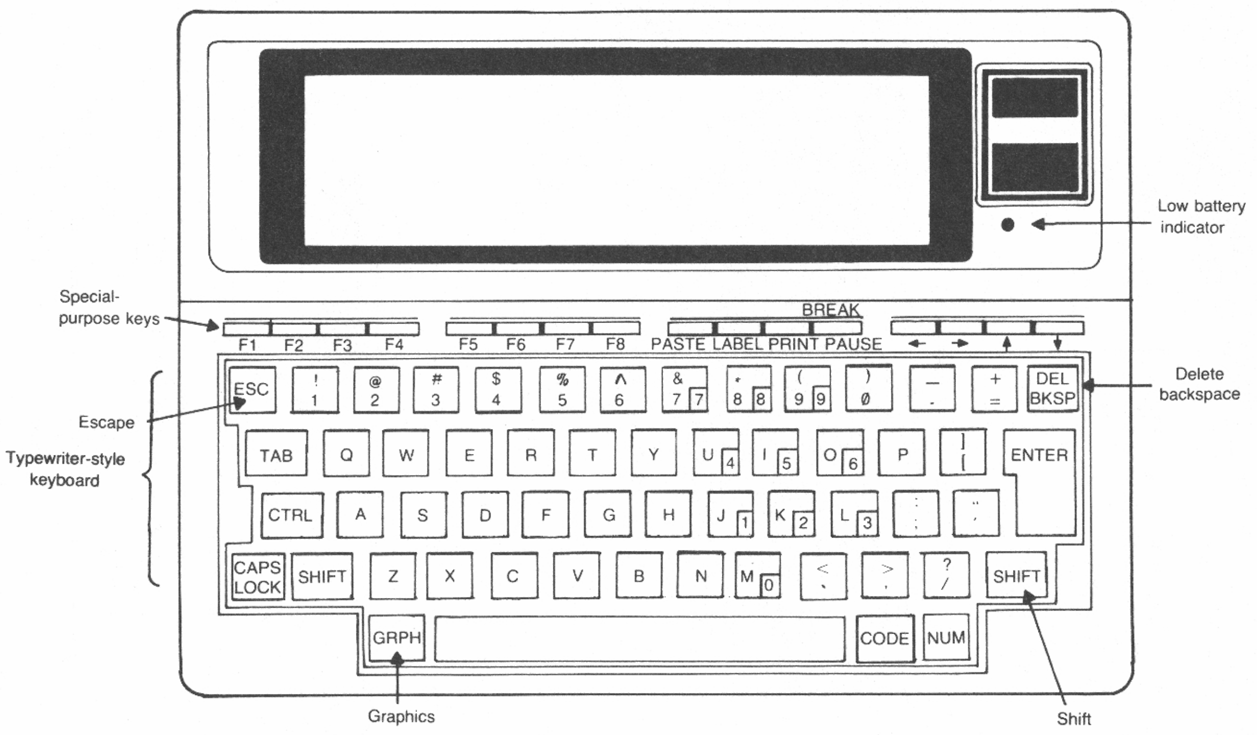


Fig. 3-3. The keyboard.

language alphabets, special symbols, and a staggering number of graphics characters.

KEYBOARD

Writing about the keyboard is a test of one's ability to use superlatives. Admittedly, the keys are a little too close together, but even so, the Model 100 has a better keyboard than some desktop computers. When designing the keyboard, Tandy sacrificed practically nothing to portability.

The action of the slightly concave keys is fast and light. As you type, the keys make a muffled clicking sound as they strike the base. It is not an absolutely quiet keyboard as some people may prefer, but the click is not loud. I like the sound.

As shown in Fig. 3-3, there are two keypads: the typewriter-style keyboard and, directly beneath the display, a row of 16 special-purpose keys. In addition, the number (NUM) key lets you use a portion of the typewriter-style keys like numeric keypad for entering long lists of numbers.

The Typewriter Keyboard

The special keys easily top Tandy's own Models I, III, and 4. There's something here for everyone. The typing keys are the standard QWERTY typewriter keyboard arrangement with the row of numeral keys and symbols across the top. There is the standard tab key, shift key, and enter key.

The typewriter keypad also includes special keys such as CAPS LOCK, DEL BKSP, NUM, GRPH, CTRL, CODE, and ESC. Here is a brief description of each.

CAPS LOCK. The Model 100 provides a caps lock key for typing uppercase nonnumeric alphabet characters. When the caps lock key is used, the numeral keys and punctuation keys work as in the unshifted position.

DEL BKSP. The delete/backspace key is very helpful. You press it to delete the character *before* the cursor. To delete the character *at* the cursor position, you press the delete/backspace key *with* the shift key.

NUM. The number key changes ten of the typing keys to a numerical keypad, the kind found on calculators.

GRPH. The graphics key enables you to type 41 graphics characters from rocket ships to diamonds. For example, if you hold down the graphics key and type Q, the Model 100 displays a stick figure.

CODE. Like the graphics key the code key enables you to type special characters. For example, if you hold down the code key and type a, the Model 100 displays ä.

CTRL. The control key enables you to type control characters and commands. For example, in TEXT, you can hold down the control key and type W to move the cursor to the beginning of the document.

ESC. As on the Model II, there is an escape key. The only use I have found so far for this key is to duplicate the F8 key to exit a document and return to the menu. It may be included for future software enhancements.

Special-Purpose Keys

Above the typewriter-style keypad, is a row of sixteen special-purpose keys including eight programmable function keys, four command keys, and four cursor movement or control keys.

F1 through F8. The programmable function keys perform different functions in each of the five built-in programs. For example, in TEXT, you press the F1 key to find a word or phrase. In BASIC,

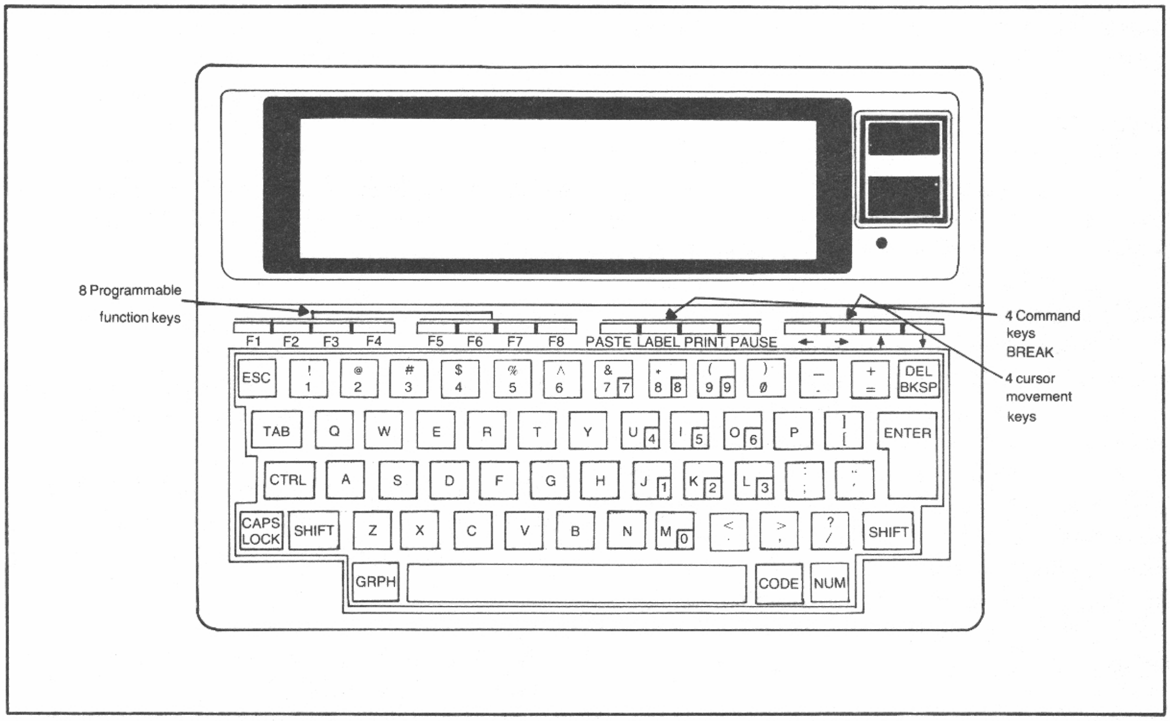

8 Programmable function keys
F1 F2 F3 F4
F5 F6 F7 F8
PASTE LABEL PRINT PAUSE
4 Command keys BREAK
4 cursor movement keys
ESC ! 1 @ 2 # 3 $ 4 % 5 ^ 6 & 7 7 * 8 8 (9 9) Ø — - + = DEL BKSP
TAB Q W E R T Y U 4 I 5 O 6 P] [ENTER
CTRL A S D F G H J 1 K 2 L 3 : ; " '
CAPS LOCK SHIFT Z X C V B N M 0 < , > . ? / SHIFT
GRPH CODE NUM

Fig. 3-4. The programmable function, command, and cursor keys.

you press the F4 key to run the program. You can use BASIC commands to program these keys yourself.

No matter what program you use, the four command keys always perform the same four jobs: paste, label, print, and break/pause.

PASTE. Press the paste key to insert text that you have previously stored. (To store the text, you use TEXT.) The paste key has a wide range of uses. For example, you can "paste" lines from one BASIC program into another.

LABEL. This is a great key. In *any* program, press the label key to list the available functions in the bottom line (line 8) of the display (Fig. 3-5). If you press the label key again, the functions disappear and line 8 is again available for your use.

PRINT. The print key enables you to print the display or an entire file.

BREAK/PAUSE. In any program, the break/pause key interrupts a function while it is in progress. Press the break/pause key to halt a program or function, or press it with the shift key to "break" execution of a program.

The Cursor Movement Keys

The last four keys are cursor movement keys.

← The left arrow key moves the cursor left.

→ The right arrow key moves the cursor right.

↑ The up arrow key moves the cursor up.

↓ The down arrow key moves the cursor down.

MEMORY

Remember folks, we are talking about a portable computer that runs on 6 volts of battery power. Tandy has perhaps come up with the most sophisticated use of the low-power CMOS (complementary metal oxide semiconductor).

Every Model 100 comes equipped with 32K (32 kilobytes) of ROM (read only memory). ROM is the preprogrammed memory that contains the operating system and the five built-in programs.

The user memory, RAM or random access memory, starts at $799 for 8K. Each additional 8K, up to a maximum of 32K, costs $120 extra. The 32K Model 100 will cost you approximately $1100.

MICROPROCESSOR

The "brain" of the Model 100 is the CMOS version of the Z80—the 8085. Its speed is 2.4 MHz. If you know how to program the Z80, you will be able to adapt many programs to the Model 100.

POWER SUPPLY

Four AA batteries or an ac adapter power the Model 100. A rechargable nickel cadmium (nicad) battery continuously feeds voltage to RAM whether the Model 100 is on or off. Even if you allow the four AA batteries to go dead, the nicad keeps the memory alive from eight to twenty hours depending on how much you have stored.

CONNECTIONS

The Model 100 is a portable with connections. You can plug in everything but the kitchen sink (that too if your sink is equipped with an RS-232-C) using the following.

dc 6V. This is where you plug in the ac adaptor. When you run the Model 100 off ac power, you save your batteries and, of course, keep the nicad battery charged. (See Fig. 3-2).

RS-232-C. This flexible serial interface (Fig. 3-6) enables you to connect the Model 100 to peripherals or to another computer. You need a standard DB-25 cable. To hook up the Model 100 to another

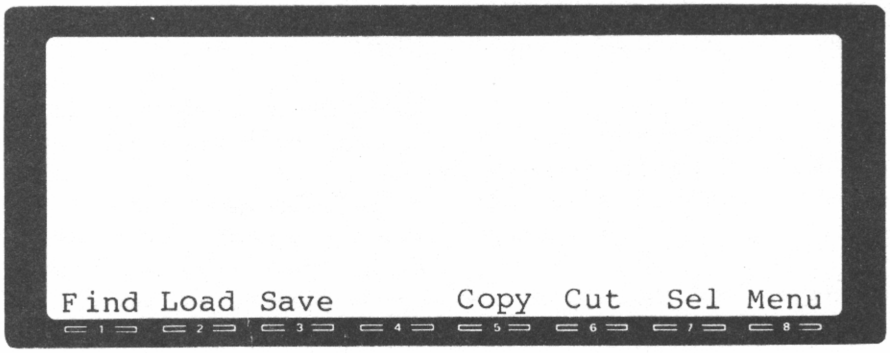


Fig. 3-5. LABEL displays the definitions of the eight function keys.

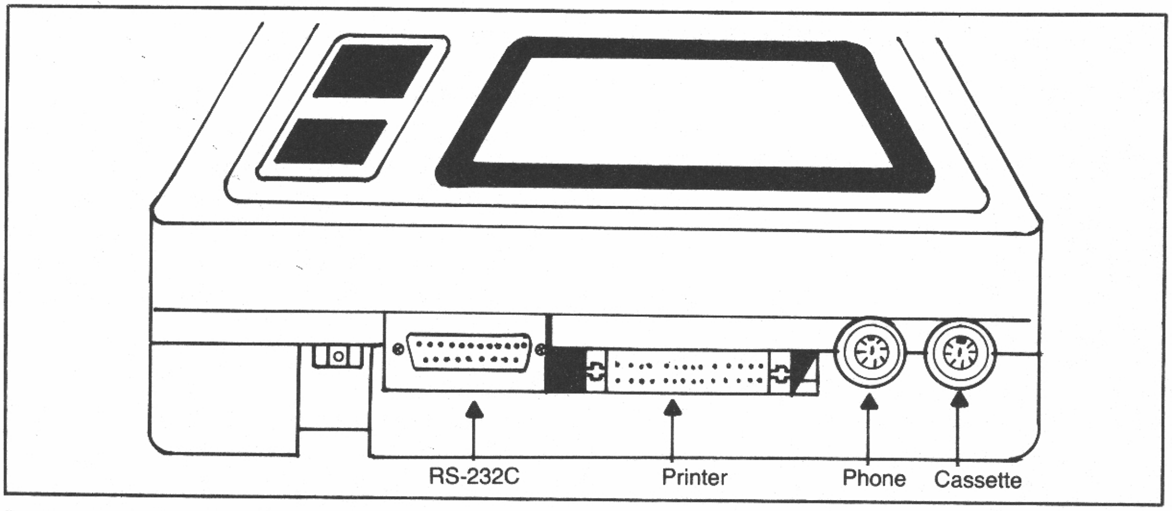


Fig. 3-6. Back view of the Model 100.

Tandy computer, you must buy an additional null modem adaptor (See Chapter 10).

Printer. A parallel printer interface (the Centronics type) enables you to hook the Model 100 to most printers (Fig. 3-6).

Direct Connect Modem (Phone). If you have modular telephone jacks, this jack (Fig. 3-6) enables you to plug the Model 100 directly into a telephone line. You must buy the Model 100 Direct Connect Modem Cable to do this. If you don't have modular telephone jacks, you can use an optional acoustic coupler.

Cassette. This is the standard Radio Shack cassette interface (Fig. 3-6) that enables you to store and load files from a standard audio cassette. The Model 100 stores at 1500 bps (bits per second).

Bar Code Wand Connector. *Bar codes* are those printed lines on products (such as kitchen cleansers) that are read by an optical device called a *bar-code wand.* This interface (BCR in Fig. 3-7) is compatible with the Hewlett-Packard wand, but right now the Model 100 does not include the necessary software. Because the connector is there, it's reasonable to assume that software for wand use will be available in the future.

MISCELLANEOUS BUTTONS, PLUGS, AND SWITCHES

To round off our tour of the hardware, let's take a look at the remaining buttons on the Model 100.

ON/OFF. Need we say more? (See Fig. 3-2).

DISP. This wheel adjusts the contrast of the LCD so you can easily view the display from any angle.

RESET. This switch (Fig. 3-2) resets the system so you can start from scratch. Most computers have a reset switch that you use when you get yourself into something that you can get out of. I never had to use the one on the Model 100.

DIR/ACP. Use this switch (Fig. 3-7) to tell the

Model 100 whether you have connected the modem directly to the telephone line (DIR) or to an acoustic coupler (ACP).

ANS/ORIG. Use this switch (Fig. 3-7) to tell the Model 100 whether you are originating a call over the modem or are answering one.

Memory. Keep your hands off this switch! When the memory switch (on the bottom of the machine) is on, the rechargable nicad battery keeps your RAM memory alive. If you turn it off, the contents of RAM are lost. This switch, must be on in order for you to operate the Model 100.

Low Battery. This indicator (See Fig. 3-2) comes on when you have approximately 15 minutes of battery power remaining. That should give you plenty of time to get to the nearest store to buy four more AA batteries.

PERIPHERALS

Here is a list of things you can plug into the Model 100:

☐ Ac Adaptor

☐ Telephone

☐ Parallel printer

☐ RS-232C cable

☐ Cassette recorder

☐ Bar-code reader

You may not want to buy every "extra" at first, but you should consider the following cables necessities when you buy your Model 100.

When you buy your Model 100 (or if you already own one), buy the ac adaptor. It is a necessity. With it, you can power the Model 100 from any standard wall outlet. The adaptor costs $19.95 and will quickly pay for itself with the money you save on batteries.

The telephone connector enables you to hook up the Model 100's modem to a modular telephone jack. Where modular jacks are unavailable, you must use an acoustic coupler to connect the computer to a telephone.

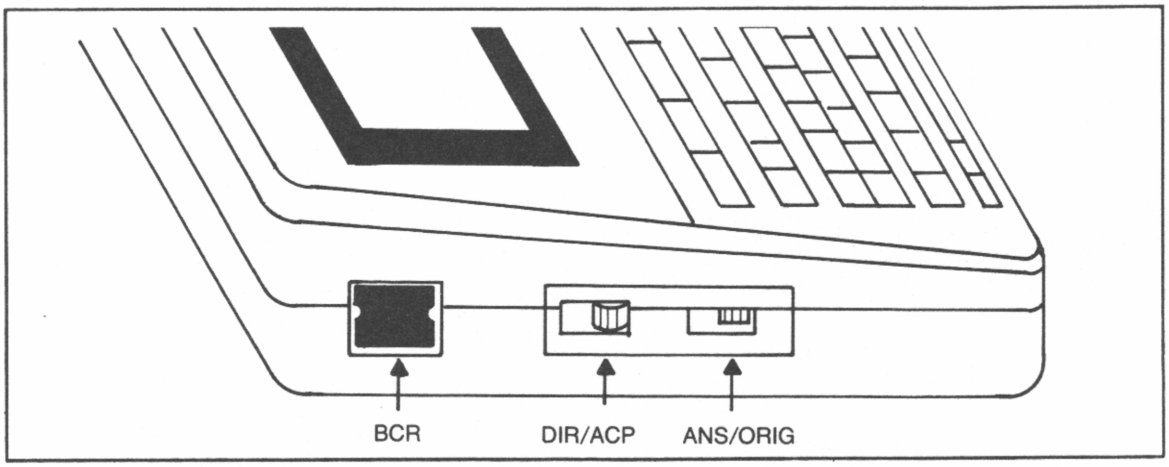


Fig. 3-7. Left side view of the Model 100.

The RS-232C cable enables the Model 100 to communicate with other computers and devices. For example, if you have a serial printer you can program the Model 100 to print through the RS-232C port rather than the parallel printer port.

If you own a printer, you'll want the Model 100 Printer Cable. It allows you to connect the Model 100 to a parallel printer (including all the Radio Shack printers). Personally, I prefer to upload my files to my Model 4, convert them to SUPERSCRIPSIT, then print them. Suffice it to say, that you can print directly from the Model 100.

If you use cassette storage, you'll want a cassette cable. You can plug a cassette cable into the Model 100 and transmit between the cassette and the Model 100 at 1500 bits per second. Personally, I prefer to upload via the RS-232C and store files on disk.

Chapter 4

Software

Every Model 100 comes with 32K ROM, packed with outstanding software designed by MICROSOFT, a leading software company. All you do is turn the Model 100 on, and the menu lists the five built-in programs: BASIC, TEXT, TELCOM, ADDRSS, and SCHEDL (Fig. 4-1). If you read about *six* built-in programs, it's because they are counting the menu as a program.

BASIC

I don't know of a better BASIC. It has all the standard BASIC statements and functions and then some. Part 5 discusses the Model 100's BASIC in detail, but here are a few standouts.

Interrupt Commands

There are five interrupt commands that branch to a subroutine under specific conditions. They are:

ON COM (communication via the RS-232C)
ON ERROR (any error)
ON KEY (any of the eight function keys)
ON MDM (communication via the modem)
ON TIME$ (at a certain time)

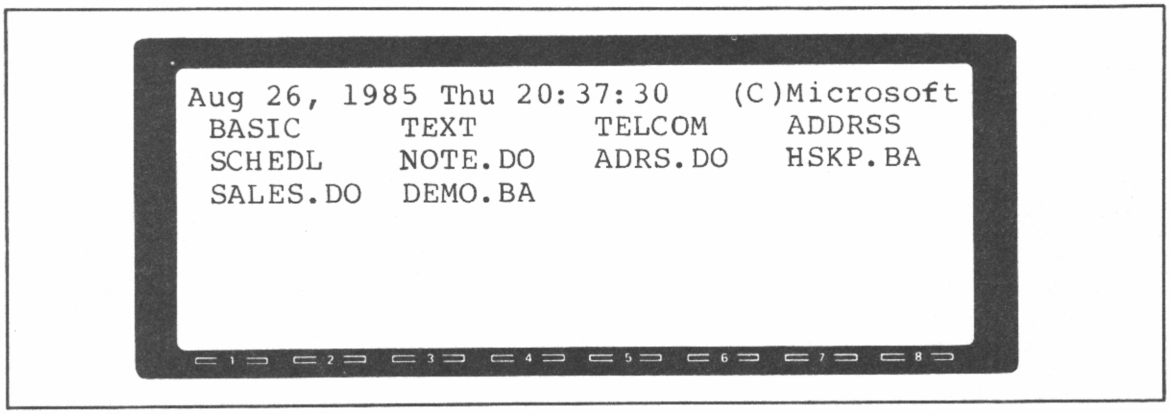
For example, if you included the following line in a BASIC program

```
ON TIME$ = "15:00:00" GOSUB 5000
```

at 3:00 PM, BASIC branches to the subroutine at line 5000. When the subroutine is complete, BASIC resumes whatever it was doing before it was interrupted. You don't have to clutter up your program with code to check the time. You can also enable, disable, and stop an interrupt.

Power Command

The Model 100 turns itself off after a few minutes if no program is running or if no keys are



```
Aug 26, 1985 Thu 20:37:30   (C)Microsoft
 BASIC      TEXT       TELCOM     ADDRSS
 SCHEDL     NOTE.DO    ADRS.DO    HSKP.BA
 SALES.DO   DEMO.BA
```

Fig. 4-1. The menu lists the five built-in programs.

pressed. The power command enables you to disable the automatic power off feature or to set the duration before it turns itself off. What's more, you can use the power command in a BASIC program to turn off the Model 100. When you turn the power back on, a resume command instructs the Model 100 to continue the BASIC program where it left off.

Sound and Lights

With the sound command you can generate tones of varying duration and pitch. With the line command you don't have to use a FOR . . . NEXT loop to draw boxes or lines. The Model 100 will automatically draw corners and fill in (set pixels) a box (Fig. 4-2).

Open Command

Another plus is the open command. OPEN enables you to treat RAM, the cassette, the RS-232C, and the modem as files for input, output, or append. OPEN also opens the printer or the display for output only. For example, you can write a BASIC program that opens three files (or buffers) (Fig. 4-3):

File 1—the modem as a file for output
File 2—the modem as a file for input

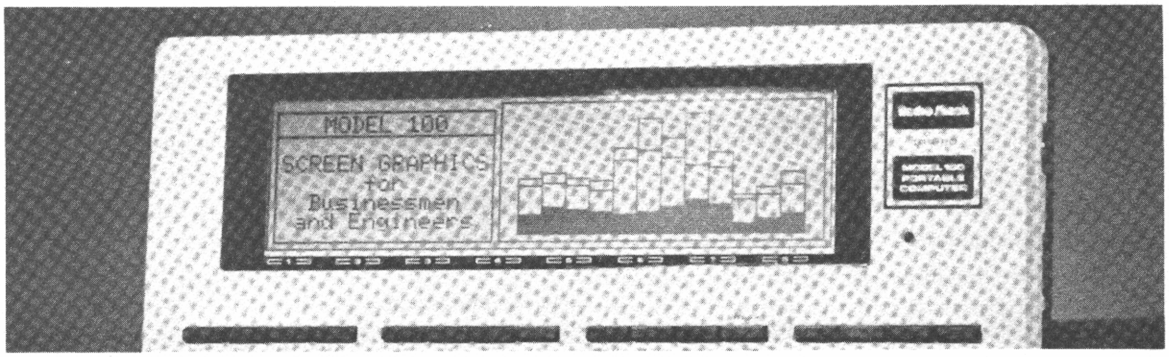


Fig. 4-2. The Model 100's graphics are remarkable for a portable.

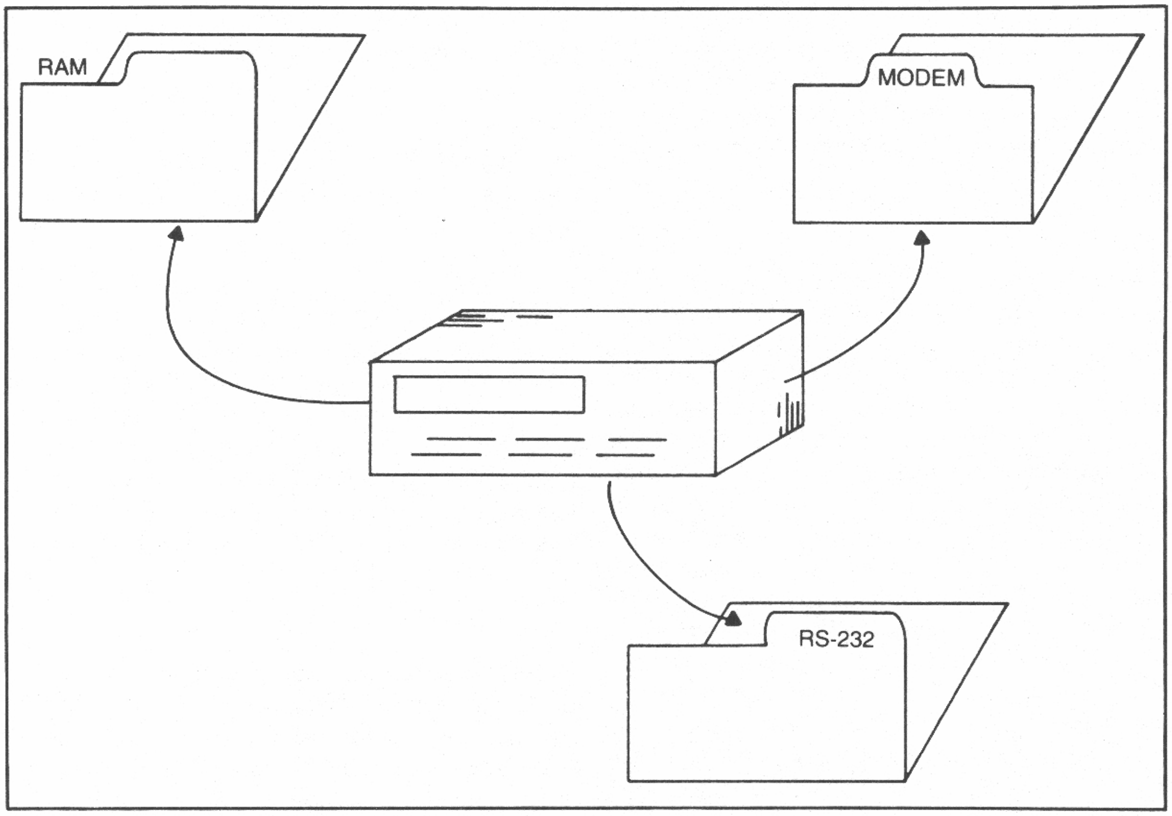


Fig. 4-3. You can write a BASIC program that opens three files; one file can serve as output from the modem, one for input through the modem, and another for RAM for output.

File 3—RAM as a file for output

The BASIC program uses file 1 to output to the modem, call a data base, and log on. It uses file 2 to input the data through the modem, and it uses file 3 to output the data to a RAM file. You can manually accomplish the same thing using the Model 100's TELCOM program.

EDIT

As I have said, one outstanding feature of the Model 100 is its integrated software. BASIC's edit command is a case in point. To edit a BASIC program, you type EDIT (you can specify a range of line numbers) and press the enter key. The Model 100 converts the program into text and puts you in the TEXT program. The full power of the Model 100's TEXT program is available for editing your program. When you finish editing you press the F8 key, and the Model 100 converts the text back into compressed BASIC code.

BASIC Interactions

BASIC is interactive with the other programs. We have mentioned how the open command enables you to use devices such as the modem and RS-

232-C in your BASIC programs. You can use BASIC to sort and edit text files. For example, the TEXT program does not include a search and replace feature. If that is something you need, you can write a BASIC program to search and replace your text files.

TEXT

For a portable, the character-based TEXT program is quite remarkable. The TEXT program is not, strictly speaking, word processing. It is, as Tandy says, "a simple but powerful text preparation tool." You cannot set margins or tabs. You cannot change formats or search and replace. You cannot print boldface characters, underscores, page numbers, or headers and footers.

Let's face it, you don't buy a portable for state-of-the-art word processing. In most cases, you use the Model 100 as a note pad, to write letters "on the road" or to prepare files for communications. If you want to snazz up a document, you can always upload your Model 100 documents to a word processor.

That said, let's take a look at some of the things that TEXT *can* do. It has all the features you need for the three main tasks of text preparation: typing, editing, and printing.

Typing

The Model 100's TEXT program shines most as a typing (text input) program. The keyboard is first rate, and the display is large enough to provide you with a workable amount of text. As you type, text automatically wraps around at the end of each line, so you don't have to worry about carriage returns.

With the caps lock key it is easy to type uppercase characters, especially when you want to intersperse numerals and capital letters. The delete/backspace (DEL BKSP) key is especially helpful for correcting typographical errors *as* you type. The graphics and control keys enable you to input special characters quickly and with ease.

Editing

The main editing features are insert, the block commands, the cursor movement commands, and find.

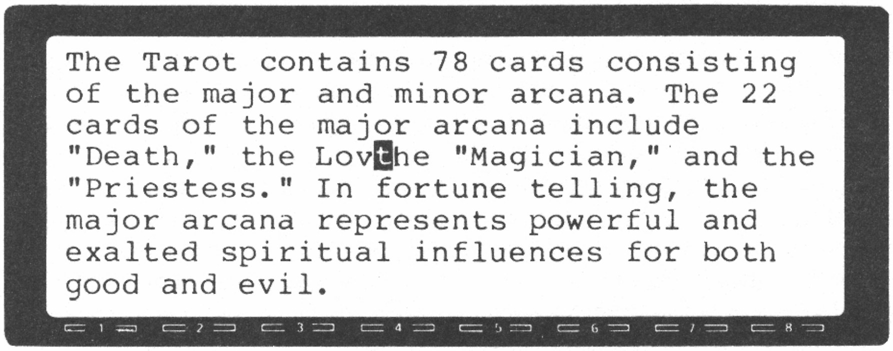
Insert. Insert is an editing feature that is both a plus and a minus: TEXT is *always* in insert mode. It's a plus because all you do to insert text is position the cursor and type (Fig. 4-4). The Model 100 automatically makes room for each character. It's a minus because you cannot fix typographical errors by typing one character over another. For example, to change the typo countru to country, you cannot type a y over the u. You must delete the u and insert the y.

Block Commands. There's nothing minus about the block commands (copy, cut, and paste) and the cursor movement commands. Let's say you want to move two paragraphs (a *block* of text) from one place to another; you use Select, Cut, and Paste.

You move the cursor to the beginning of the block, press the F7 key to choose Select, then move the cursor to the end of the block. The cursor movement commands make it easy to select a quantity of text. As you select the block, the Model 100 highlights it in reverse video, white letters on a black background (Fig. 4-5). You press the F6 key to choose Cut. The Model 100 will delete the block from the file and store it in a temporary buffer.

You position the cursor where you want to insert the cut block and press the paste key. The Model 100 inserts (pastes) the block at the cursor position.

Besides cutting a block of text, you can also copy it. Whether you copy or cut, the paste key enables you to insert the block anywhere—in a



```
The Tarot contains 78 cards consisting
of the major and minor arcana. The 22
cards of the major arcana include
"Death," the Lovthe "Magician," and the
"Priestess." In fortune telling, the
major arcana represents powerful and
exalted spiritual influences for both
good and evil.
```

Fig. 4-4. To insert, you just position the cursor and type.

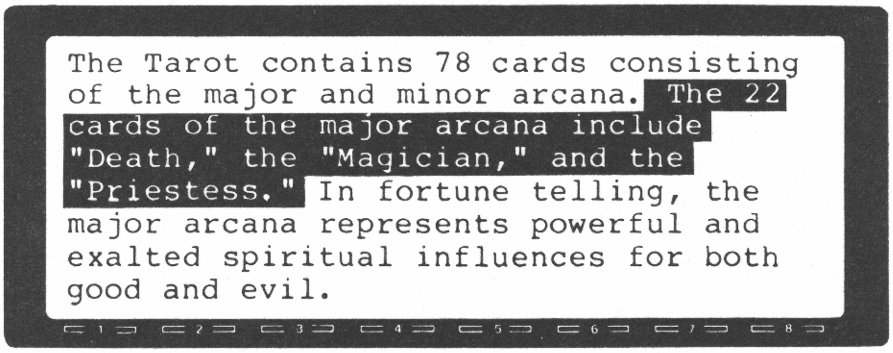
different document, even in a BASIC program.

Finally, the Model 100 offers a surprisingly quick find command. With it you can find a string of up to 24 characters.

Printing Text

Most of what TEXT fails to offer as a word-processing program involves printing. Printing in itself is easy, it's just that the Model 100 offers no "bells and whistles." (I'll say it again. I don't think word-processing bells and whistles are all that desirable in a portable.)

The Model 100 provides a Centronics-type port to connect it to a parallel printer. The print command key prints the current contents of the display. Pressing the shift and print keys prints the entire file. Before it prints, TEXT asks you to set the width of the printout. However, TEXT does not provide features such as left margin, line spacing, page numbering, pagination, and so on. There are two ways to get around some of these deficiencies. You can upload your text files to another system that offers full word processing, or you can write a BASIC program to print your text files using the features you want.



```
The Tarot contains 78 cards consisting
of the major and minor arcana. The 22
cards of the major arcana include
"Death," the "Magician," and the
"Priestess." In fortune telling, the
major arcana represents powerful and
exalted spiritual influences for both
good and evil.
```

Fig. 4-5. You can select "blocks" of text to delete or copy.

SCHEDL AND ADDRSS

I've grouped SCHEDL (Schedule) and ADDRSS (Address) together because their functions are quite similar. They are essentially "souped up" data-base managers that enable you to find pretyped information quite rapidly. In addition Address has additional uses in conjunction with the TELCOM program.

Both programs are perfect examples of software designed specifically for a portable computer. Between them, it's possible that your appointment book, expense record, and address book could join the ranks of the obsolete. Faster than you can get out your address book, find a name, and dial the number, the Model 100 can do all that for you. And the files that list names and appointments are much easier to update than are an address book or calendar. (Does your address book look like mine—friends and relatives who move so often that by now their phone number is squeezed somewhere in the margin?)

Let's use the ADDRSS program as an example. Before you use ADDRSS, you must use TEXT to type and store a list of names, addresses, and phone numbers. You name this file ADRS.DO.

Typing the ADRS.DO File

For each item in your list, you will probably include name, address, telephone number, description, and any other data you might need.

Tandy suggests that you preface each item with a unique identification. For example, if you run a large corporation, you might type !V before the name of each Vice President, !D for each department head, !S for supervisors, and so on.

In my more humble ADRS file, I use !F for family members, !B for business associates, and !P for personal friends. My ADRS.DO file looks something like Fig. 4-6. So that I can use TELCOM to dial numbers, each phone number is enclosed in colons.

Using ADDRSS

The ADDRSS program enables you to search the ADRS.DO data base in anyway. You press the F1 key for Find, and type any group of characters. For example, by finding !B, the Model 100 displays all my business associates. By finding !F, it displays all my family members. You can search by name or any part of a name. If you press the F1 key and don't type anything, the Model 100 displays all items a display-full at a time. You can also use LFIND (the F5 key) to print items.

And of course, because TEXT makes revising your ADRS file so easy, you can quickly and easily add, delete, or change items.

As you will read below, the Model 100 also uses the ADRS data base with TELCOM.

```
!B Stuart Gibson, Concept Industries, 1116 Summer Street,
Stamford, Connecticut, 06905  :5550982:
!F Mom and Dad, 300 Home Street, Greensboro, N.C.  27400
:9192997007:
!P Mimi Larue, 618 21st Street, NY, NY, 10014 :6829000:
```

Fig. 4-6. A sample ADDRSS document.

```
$    6/8/83     34.00      Roundtrip to Hartford.
$    6/8/83     15.00      RS-232 Cabel
P    6/21/83    2:15       Dentist
B    6/15/83    9:00       Chemical Bank Meeting
P    6/14/83    10:15      Take cats to Vet
!    6/27/83               Model 100 Book due at TAB
!    8/31/83               4 lessons due for Chem Bank
```

Fig. 4-7. A sample SCHEDL document.

A Few Words about SCHEDL

SCHEDL works the same way as ADDRSS, only you use TEXT to create a document named NOTE.DO.

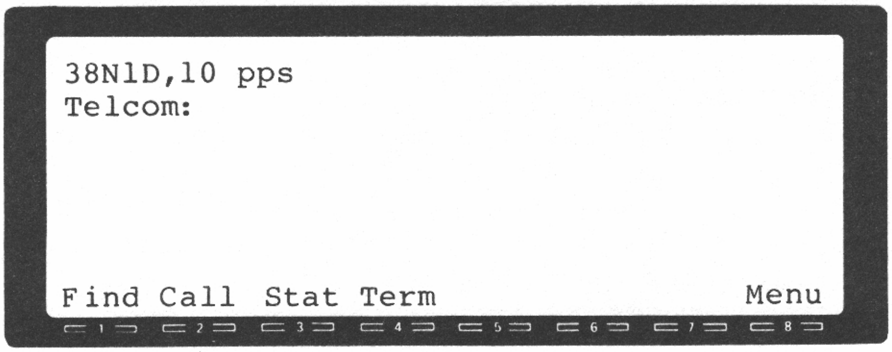
Instead of names and addresses, you type appointments, deadlines, expense account items, etc. You use identifications for your NOTE.DO file also. For example, I use SCHEDL to keep track of manuscript deadlines, travel expenses, and personal and business appointments. I preface each deadline with an exclamation point (!), each expense account item with a dollar sign ($), each business appointment with a letter B and each personal appointment with a letter P. My NOTE.DO file looks something like Fig. 4-7.

By finding $, SCHEDL lists my expenses. I can find data by date (year, month, or day), by time, or by description. For example to find out when the cats are getting their shots, I could FIND !, 10:15, 6, 6/14, cat, Vet, etc.

TELCOM

The TELCOM program makes the Model 100 a super communicator. By hooking up the Model 100's modem to a telephone, you can use it as an automatic dialer. You can instruct the Model 100 to dial a data base (such as CompuServe), log on, request information, and download the data into RAM. You can also use TELCOM with the RS-232C interface to communicate with another computer or a peripheral.

To give you a rough idea of what TELCOM is,



```
38N1D,10 pps
Telcom:
```

Fig. 4-8. The function key definitions in TELCOM's entry mode.

here is a brief and incomplete summary of the main commands. TELCOM provides two modes: entry and terminal.

Entry Mode

When you select TELCOM from the menu, you enter TELCOM's entry mode (Fig. 4-8). You use this mode, among other things to set communications parameters and to autodial.

The STAT (status) command enables you to set a wide range of communications parameters such as baud rates, word lengths, parities, etc.

The find command enables you to locate a phone number in the ADRS.DO file. The call command dials numbers automatically. You also use FIND and CALL to log on automatically to a data base. The term command turns on terminal mode.

Terminal Mode

You must be in terminal mode (Fig. 4-9) to communicate (either through the modem or the RS-232C) with another computer. Here are the commands available.

PREV. Previous (F1). Press F1 to display the previous eight lines.

DOWN. Download (F2). Press F2 to download the data into a RAM file as it comes in from the modem or RS-232-C.

UP. Upload (F3). Press F3 to upload (send) data from one of the Model 100's RAM files through the modem or RS-232-C.

FULL. Full or half duplex (F4). Press F4 to switch back and forth between full and half duplex.

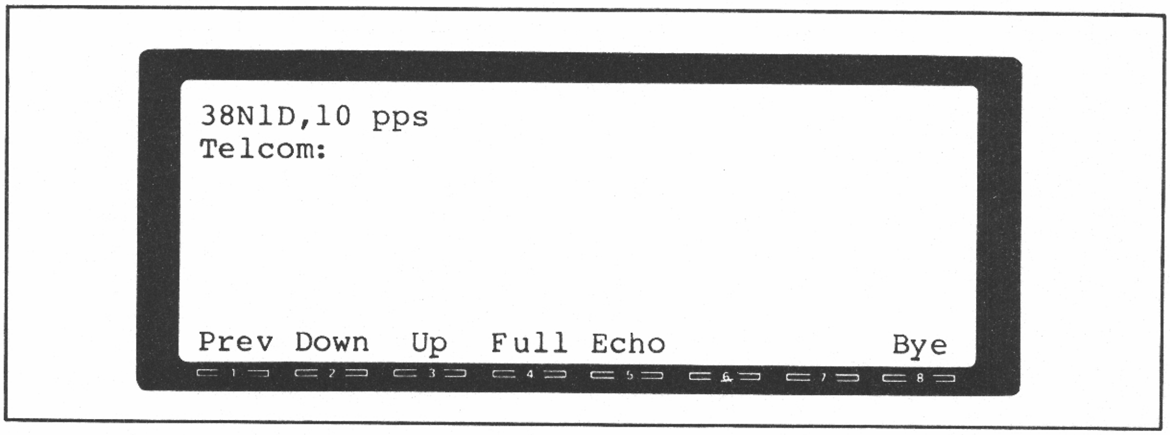
ECHO. (F5) Prints data as it comes in.

BYE. Disconnect (F8). Press F8 to disconnect from the host computer.

For example, auto dial is perhaps the simplest TELCOM feature to understand and master. Let's call my personal friend, Mimi Larue, whose number is stored in the ADRS.DO file. From the menu you select TELCOM, and press F1 to find "Mimi." The Model 100 displays Mimi's record. Press F2 to call the number. The Model 100 dials Mimi's number and displays each number as it dials.

Is Anything Wrong?

It might surprise you to learn that TELCOM is



```
38N1D,10 pps
Telcom:





Prev Down  Up   Full Echo           Bye
```

Fig. 4-9. The function key definitions in TELCOM's terminal mode.

almost but not perfect. You cannot use the modem and the RS-232-C at the same time. You cannot use TELCOM to use the Model 100 as a modem by another computer.

To take data over the phone for use by another computer, you must do it in steps. First log on through the modem, download the data into a RAM file as it comes in. Then use TEXT to edit the data.

Next, use TELCOM mode to change the communications settings from modem to RS-232-C. Then use terminal mode to upload the RAM file to the other computer.

As usual, you can get around all this by writing your own BASIC program.

PART 2
TEXT

This part is about the TEXT program. In Chapters 5 through 8 I'll describe how the Model 100 handles the main tasks of text preparation: typing, revising, printing, and file management. I'll also include the BASIC commands that you must know to use TEXT effectively. Finally, in Chapter 8, you will find hands-on exercises that will enable you to put TEXT through its paces.

You use text with all the other programs, for example, use text with

SCHEDL	to prepare the NOTE.DO data base.
ADDRSS	to prepare the ADRS.DO data base.
BASIC	to edit BASIC programs.
TELCOM	when you download or upload files.

This Part focuses on the TEXT program, but because the Model 100's various programs are so interactive, there are many additional ways to work with TEXT. You can write a BASIC program to search and replace in a document or to print your documents with various line spacings, page numbers, headers and footers, and so on. You can use either BASIC or TELCOM to upload your documents to another computer or word processor, or to print through the RS-232. Part 5 provides ideas for using BASIC with TEXT, and Part 4 provides ideas for using TELCOM with TEXT.

Chapter 5

Typing with TEXT

Before we get rolling, we must come to grips with two potentially confusing terms: file and document.

RAM FILES

The Model 100's software provides access to RAM (user memory) via *files*. If you are used to cassette or disk storage, then you know how to write to (output to) a file and to read (input from) a file. On the Model 100 you read and write RAM files. You store BASIC programs and TEXT documents in RAM as files. For example, if you save a BASIC program and name it SORT, the Model 100 adds .BA to the program name, stores it in RAM, and displays the file name in the menu as SORT.BA. Figure 5-1 lists several BASIC program files.

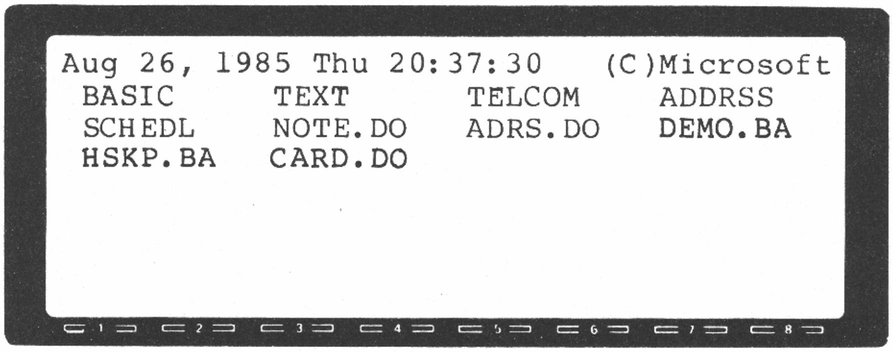
When you use the TEXT program to type a document, the Model 100 requests a file name. Let's say you name your document MEMO. As you type, the Model 100 stores each character in the RAM document file rather than in a BASIC file. In the menu, the Model 100 displays the document file name (with the extension .DO) as MEMO.DO. Figure 5-1 lists several document files.

So, in TEXT you type, edit, and print *documents*. The Model 100 stores your documents in RAM as document files. Document files are stored in ASCII code format. (If ASCII is new to you, you'll find it described in Chapter 12.)

A document file can be as short as one character (although that would be highly unusual), or as long as the amount of memory allows. As a rule of thumb, each character uses up one byte.

SELECTING TEXT

As with any program on the Model 100, you start from the menu. There are two ways to select the TEXT program. You can position the cursor on TEXT and press the enter key. Then type the document name and press the enter key again. Alternatively you can position the cursor on the



```
Aug 26, 1985 Thu 20:37:30   (C)Microsoft
 BASIC      TEXT      TELCOM    ADDRSS
 SCHEDL     NOTE.DO   ADRS.DO   DEMO.BA
 HSKP.BA    CARD.DO
```

Fig. 5-1. The Model 100 adds .BA to programs, .DO to documents.

document name (such as MARY.DO) and press the enter key.

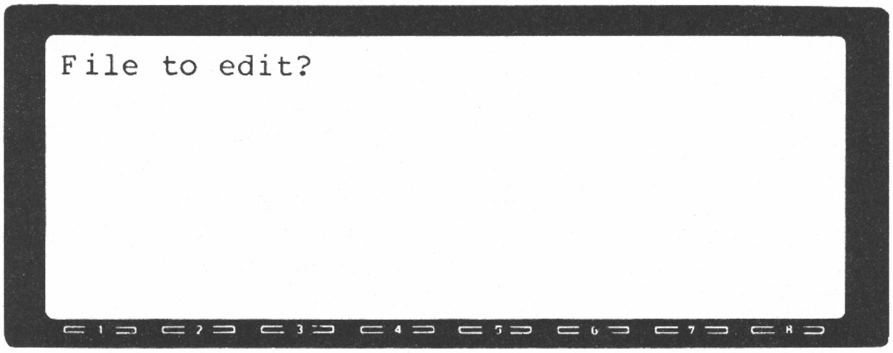
TYPING A DOCUMENT

To type a new document on the Model 100, you select TEXT and type a name for your new document. This is called *opening a file*. There really is nothing to it. On the menu, position the cursor on TEXT and press the enter key. The Model 100 clears the screen and prompts: File to edit? (Fig. 5-2).

Now there is a *little* something to it. You must type a name for the document so the Model 100 can store it in RAM. Then press enter again.

You can type any combination of letters and numerals as the file name as long as the document name is no more than six characters and the first character is not a numeral.

You could name a file MEMO, but you could not name one MEMORANDUM. There are more than six characters in MEMORANDUM, and there isn't enough room on the menu for it. You could name a file R2D2, but you could not name one 2MUCH because the first character is a numeral.



```
File to edit?
```

Fig. 5-2. When you select TEXT, you are asked for a file name.

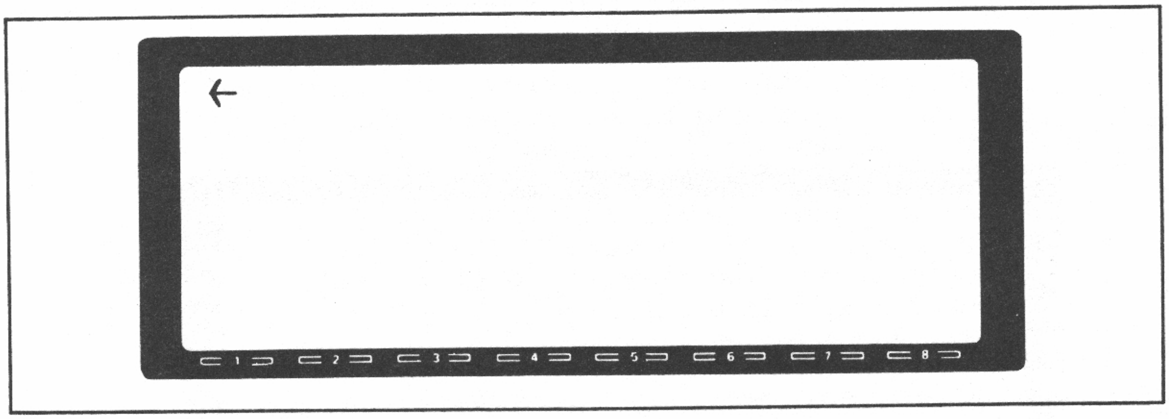


Fig. 5-3. If you open a new file, a screen with an arrow in the top left-hand corner is displayed.

Before you press the enter key to complete the response, you can use the delete/backspace key to change the name. After you type a name and press the enter key, the Model 100 displays a blank screen.

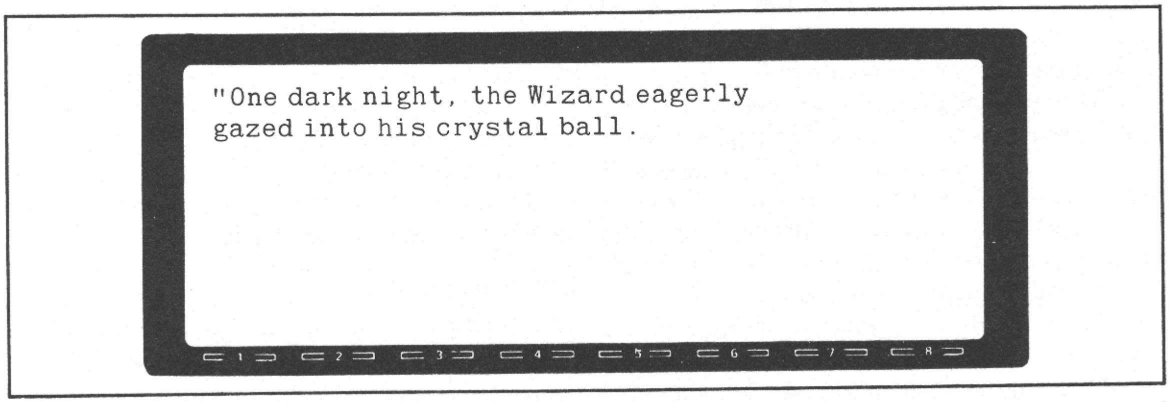
At the first character position (the top left-hand corner), the Model 100 displays a left pointing arrow: ← (Fig. 5-3). It always appears at the very end of a document so that you will know where the end is.

All you do now is type. The Model 100 displays each character as you type it.

WRAPAROUND, CARRIAGE RETURNS, AND SCROLLING

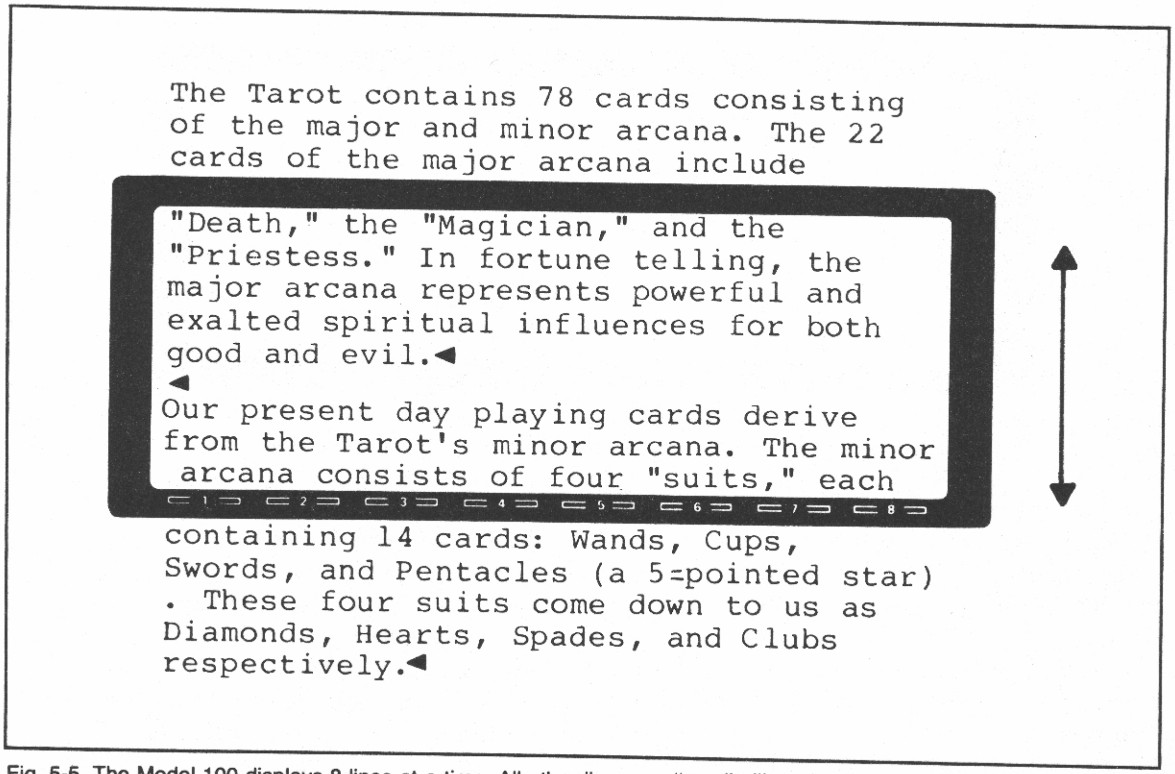
I'm going to talk about wraparound, carriage returns, and scrolling now. If you already know about these word-processing conventions, you may want to skip ahead to Key Typing Hints.

One of the most helpful TEXT features is *wraparound*. Unlike typing on a typewriter, you never have to think about starting a new line when you near the right margin. Because the screen can only display 40 characters on a line, if you typed, "One dark night, the Wizard eagerly gazed into



```
"One dark night, the Wizard eagerly
gazed into his crystal ball.
```

Fig. 5-4. The Model 100 can display only 40 characters (including spaces) on a line. If the fortieth character falls in the middle of a word, the machine automatically moves the entire word to the next line, a feature called wraparound.



```
The Tarot contains 78 cards consisting
of the major and minor arcana. The 22
cards of the major arcana include
"Death," the "Magician," and the
"Priestess." In fortune telling, the
major arcana represents powerful and
exalted spiritual influences for both
good and evil.◄
◄
Our present day playing cards derive
from the Tarot's minor arcana. The minor
 arcana consists of four "suits," each
containing 14 cards: Wands, Cups,
Swords, and Pentacles (a 5=pointed star)
. These four suits come down to us as
Diamonds, Hearts, Spades, and Clubs
respectively.◄
```

Fig. 5-5. The Model 100 displays 8 lines at a time. All other lines are "scrolled" or cleared from the screen.

his crystal ball.", the entire word *gazed* would not fit on the line. The *d* in *gazed* is the 41st character. Keep typing anyway. The model 100's wraparound feature will automatically move the entire word *gazed* to the next line as shown in Fig. 5-4.

When you want to end a line before the right margin, you press the enter key to type a *carriage return*, and the next words to be displayed will begin on a new line. (On a typewriter a carriage return is when the pattern rolls the paper up a line and returns the printhead or carriage back to the left margin ready to start a new line.)

When you press the enter key at the end of a line, the model 100 displays a◄(sort of carriage return) and moves the cursor to the left margin of the next line. You also press the enter key to end a paragraph or to create a line space.

The display has room for eight lines of text. When you begin typing the ninth line of a document, the first line moves up and off the display and line nine appears at the bottom of the display. If you move back up to display line one, line nine moves down and off the bottom of the display (Fig. 5-5). This is called *scrolling*. In Chapter 6 on revising we'll talk about the commands you use to scroll text up and down through the display.

KEY TYPING HINTS

The Model 100 is a great machine for typing. Not only is the keyboard excellent, but it also includes some great keys that vastly simplify typing chores. These are TAB, SHIFT, CAPS LOCK, CODE, GRPH, NUM, DEL BKSP. (The cursor keys and function keys are discussed in Chapter 3.)

These following keys (Fig. 5-6) are found on almost all computers or word processors.

ENTER. You press the enter key to begin a new line, to end a paragraph, or to type a line space.

TAB. Because of the small display, TEXT's tab capability is very limited. The tab stops are preset every 8 characters: 8, 16, 24, and 32. When you press the tab key, the cursor jumps to the next tab stop. For example if you have typed up to character 22 and you press the tab key. The cursor jumps to character position 24.

SHIFT. There's nothing new here. Hold down the shift key to type uppercase characters or to type the shift symbols on top of the numeral and punctuation keys.

These following keys enable you to type special characters and capital letters (Fig. 5-6).

CAPS LOCK. Now we're getting somewhere. Unlike SHIFT, CAPS LOCK only affects the alphabet keys. When you press the caps lock key, it stays down and displays each alphabet character you type in uppercase. When you press a numeral key or punctuation key, however, numbers and punctuation marks can be typed as in the unshifted position.

Usually, you use CAPS LOCK to type text that combines capital letters with numbers, as in

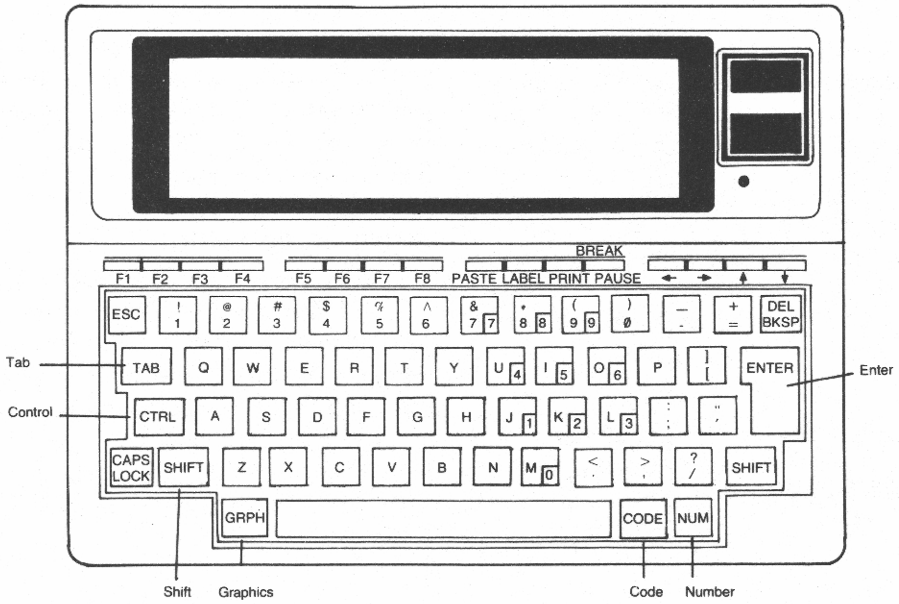
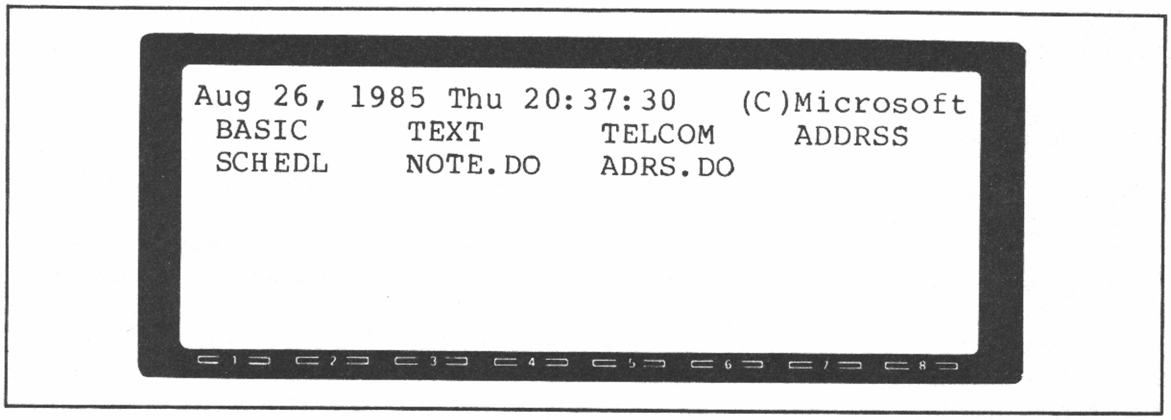


Fig. 5-6. The enter, tab, shift keys, and the special-character graphics, code, number, and control keys.



```
Aug 26, 1985 Thu 20:37:30    (C)Microsoft
 BASIC      TEXT       TELCOM     ADDRSS
 SCHEDL     NOTE.DO    ADRS.DO
```

Fig. 5-7. When you press the F8 key, you exit the document, store it in RAM, and return to the menu.

WIDGETS SOLD	2,456,120
TOTAL PRICE	$1,345.60

Press CAPS LOCK again to release it.

CODE. You hold down the code key and press another key to type special characters. For example, if you hold down CODE and type A, the Model 100 displays ä. If you hold down the Shift and Code keys together and type A, the display will be Ä.

GRPH. Graphics. When you hold down the graphics key and press another key, the display shows one of 41 graphics characters. For example, if you hold down GRPH and type Q, the display will be a stick figure.

NUM. The number key changes the function of ten typing keys (the ones with the numerals in little white squares (in their lower right-hand corners) so that they function as a numerical keypad, the kind found on calculators.

DEL/BKSP. The delete/backspace key is great for fixing typographical errors as you type. Press this key to delete the character you just typed. For example, type Wizarf. The cursor will appear after the letter *f* waiting for the next character. By pressing the delete/backspace key, you move the cursor directly over the letter *f*, deleting that letter. You then type d to make Wizard.

EXITING THE DOCUMENT

As you type, the Model 100 stores your text in the RAM document file. When you finish typing, you can proofread and revise the document then and there, or you can exit the document, return to the menu, and revise the document at a later date.

To exit a document and close the file, you press F8 for menu (Fig. 5-7). The Model 100 displays the menu and lists your new document.

Chapter 6

Revising and Printing with TEXT

The Model 100 is a great little text editor. Of course, in order to revise a document, you first display it. To display a document, return to the menu, position the cursor on the document name, and press the enter key.

TEXT provides:

- ☐ All the cursor movement commands you could want. To revise text, you must first position the cursor at the point of change.
- ☐ Two ways to revise: character by character and by block of text.
- ☐ A very fast find command to locate a word or phrase.

LABEL

Before I get into specifics, let me dispense with the label key. When you are in TEXT, you can display the definitions of the function keys by pressing the label key (Fig. 6-1). When you do this, you temporarily lose the eighth line of the display.

To clear the list of function key functions and regain the eighth line, just press the label key again.

CURSOR MOVEMENT COMMANDS

To move the cursor through TEXT, you use the four cursor keys alone or with either the shift or the control key (Fig. 6-2). The cursor keys repeat. Whether you use them alone or with the shift key, you can hold them down to continuously move the cursor. Here is how they work.

To move the cursor one character or line at a time, you use the cursor keys by themselves.

← Moves the cursor one character to the left.

→ Moves the cursor one character to the right.

↑ Moves the cursor up a line.

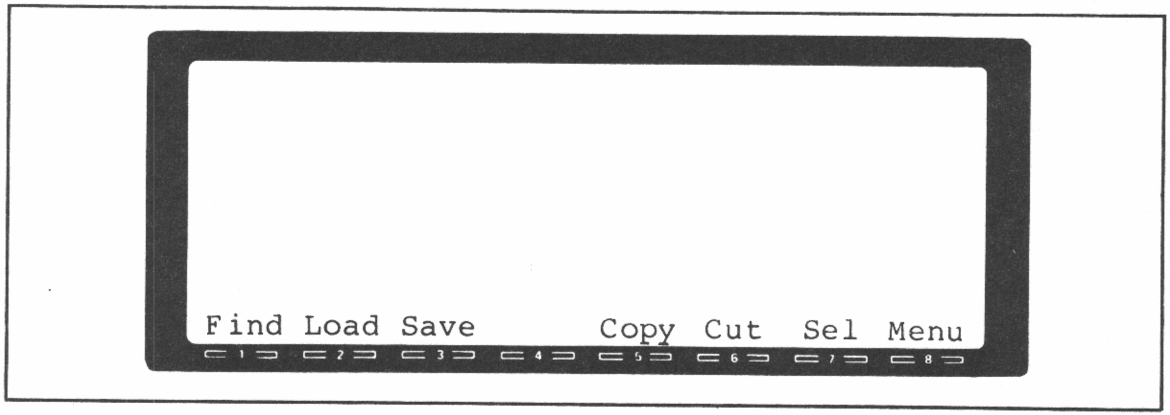


Fig. 6-1. The function keys in TEXT.

↓ Moves the cursor down line.

To move the cursor a "screenful" or word at a time, you hold down the shift key and press a cursor key.

Shift → Moves the cursor to the next word.

Shift ← Moves the cursor to the previous word.

Shift ↓ Moves the cursor down eight lines to the next screenful. The eight lines on the display scroll up.

Shift ↑ Moves the cursor up eight lines to the previous screenful. The eight lines on the display scroll down.

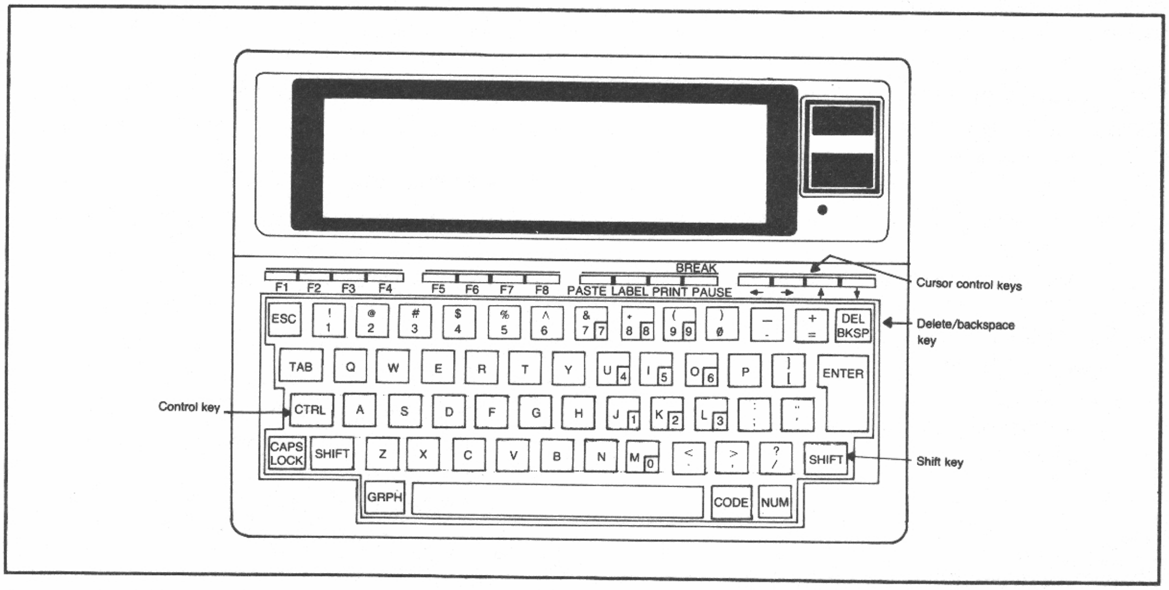
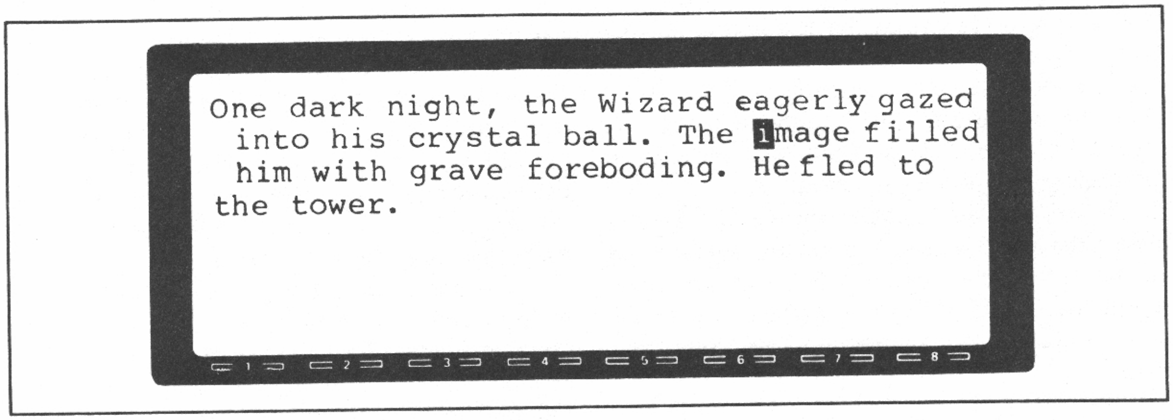


Fig. 6-2. The four cursor and shift and control keys.



```
One dark night, the Wizard eagerly gazed
 into his crystal ball. The image filled
 him with grave foreboding. He fled to
the tower.
```

Fig. 6-3. The cursor position for inserting the word *gruesome*.

To move the cursor to the outer limits—that is to the beginning or end of a line or document—you hold down the control key and press a cursor key.

CTRL → Moves the cursor to the end of the line.
CTRL ← Moves the cursor to the beginning of the line.
CTRL ↑ Moves the cursor to the beginning of the document.
CTRL ↓ Moves the cursor to the end of the document.

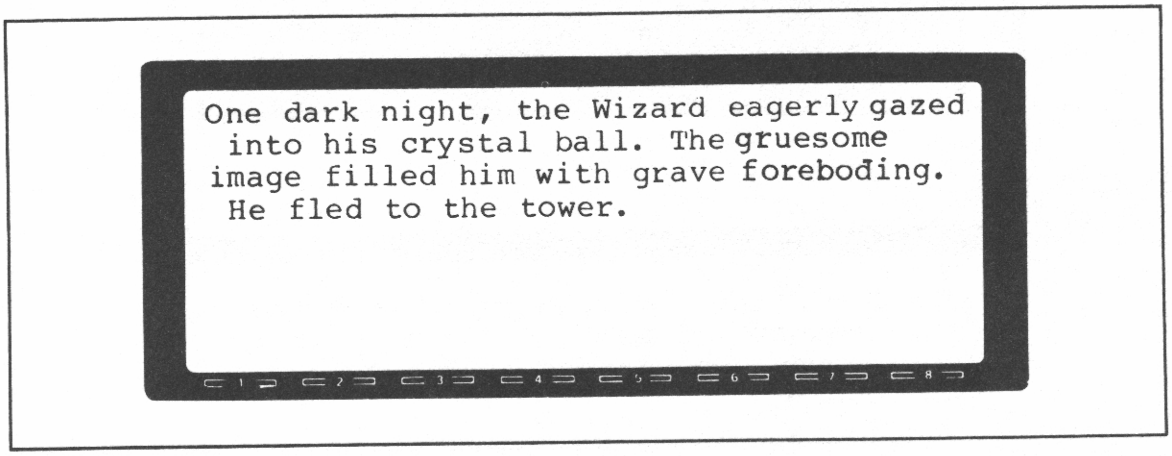
REVISING

There are two ways to revise text: character by character or by block of text.

Revising by Character

TEXT enables you to revise by character, that is you can insert or delete one character at a time.

Inserting Characters. TEXT is always in the mood to insert because it's always in insert mode. If you position the cursor on a character and type, the



```
One dark night, the Wizard eagerly gazed
 into his crystal ball. The gruesome
image filled him with grave foreboding.
 He fled to the tower.
```

Fig. 6-4. The screen after the insertion has been made.

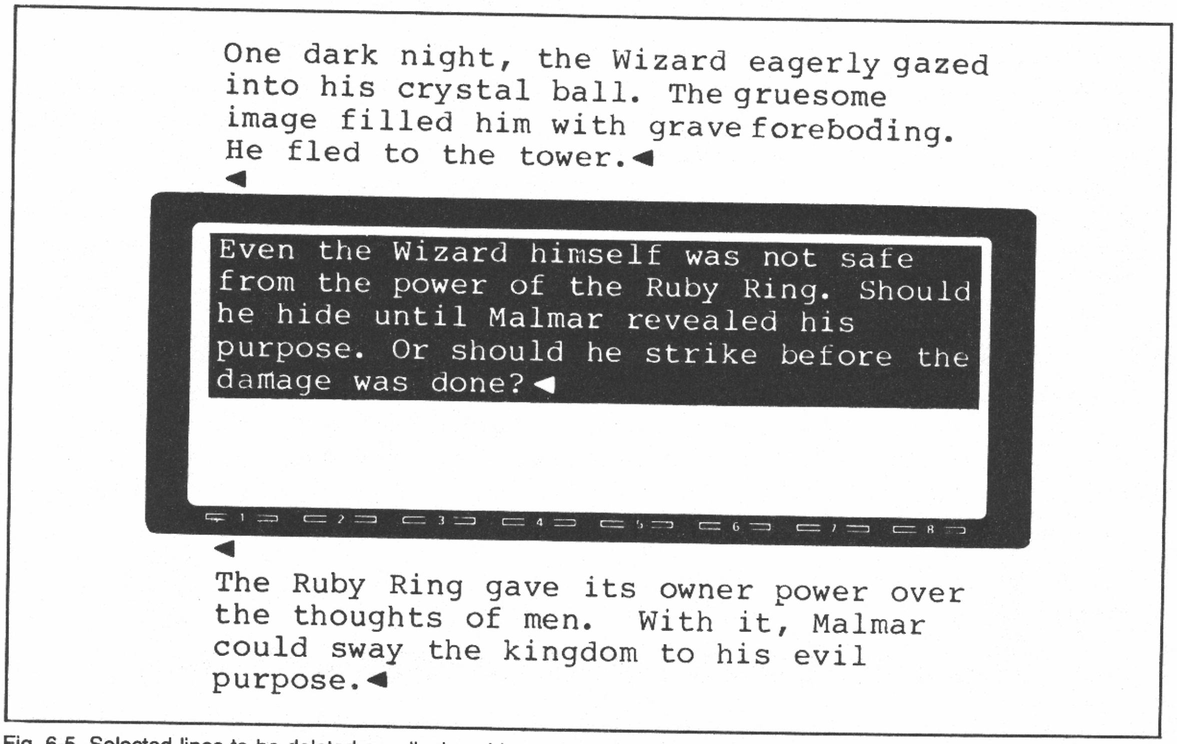
Model 100 makes room for the new text by moving everything after the cursor over and down.

For example, let's say that you have typed **One dark night the Wizard eagerly gazed into his crystal ball. The image filled him with grave foreboding. He fled to the tower.◄**

If you wanted to insert the word *gruesome* before the word *image,* you simply position the cursor on the *i* of image (Fig. 6-3) and type **gruesome.** The Model 100 moves all the text over and down to make room for the new characters (Fig. 6-4).

Deleting Characters. You use the delete/backspace key (Fig. 6-5), alone or with the shift key, to delete one character at a time. You generally use the delete/backspace key in the unshifted position when you are typing because it deletes the character to the left of the cursor. For example, if you typed **One daark night, the Wizard eagerly,** you could easily delete the extra *a* in *dark.* First, position the cursor on the *r* in *dark,* then press the delete/backspace key. The cursor will backspace and delete the *a.* Your text would then read **One dark night, the Wizard eagerly.**

You generally use the delete/backspace key with the shift key when you are revising text because it deletes the character at the cursor. For example, to correct **One daark night, the Wizard eagerly,** position the cursor on the extra letter *a,* hold down the shift key, and press the delete/backspace key. The Model 100 will delete the extra *a,* and your copy would read **One dark night, the Wizard eagerly.**



```
One dark night, the Wizard eagerly gazed
into his crystal ball. The gruesome
image filled him with grave foreboding.
He fled to the tower.◄
◄
Even the Wizard himself was not safe
from the power of the Ruby Ring. Should
he hide until Malmar revealed his
purpose. Or should he strike before the
damage was done?◄
◄
The Ruby Ring gave its owner power over
the thoughts of men.  With it, Malmar
could sway the kingdom to his evil
purpose.◄
```

Fig. 6-5. Selected lines to be deleted are displayed in reverse video, white characters on a black background.

Revising by Block

This is the real revising power of TEXT. Three function keys—F5, F6, and F7 (Fig. 6-1)—enable you to select any amount of text as a block, then copy the block or cut it. Whether you cut or copy, the Model 100 saves the text in a temporary memory location called a *buffer.* You can then insert the cut or copied text by pressing the paste command key.

Selecting a Block. The steps are simple. Position the cursor at the beginning of the text you want to select, press the F7 key, and move the cursor to the end of the text you want to select.

For example, let's say you want to delete the second paragraph of text shown in Fig. 6-5. First, position the cursor on the *E* of *Even*, and press the F7 key for Select. Then, press the down arrow control key five times to select each line as part of the block. Each time you press the down arrow key, the Model 100 highlights the text you have selected by displaying it in reverse video (Fig. 6-5). You can add or eliminate copy from the block by moving the cursor back up or to the left.

Copying or Cutting the Block. Now you can press the F5 key to copy the block, or you can press the F6 to cut it. If you copy it, the block remains in your text, but the Model 100 also stores it in the paste buffer. If you cut it, the Model 100 deletes the block from the document and stores it in the paste buffer. So, whether you copy or cut, the block is saved in the paste buffer.

Pasting the Block. Once you have selected, then copied or cut a block, you can *paste* it anywhere: in the same document, in a different document, even in a BASIC program. This is useful for editing programs.

To paste a block, position the cursor where you want to insert the block, and press the paste key.

The paste buffer holds one block at a time. For example, when you cut a block, the Model 100 stores it in the paste buffer. If later, you copy or cut another block, the new block replaces the old block in the buffer.

FIND

Our final editing feature is the find key. Once you have displayed a document, the find key /F1/ enables you to locate a word or a phrase (called a *string* with amazing speed.

Press F1. The Model 100 prompts you with String: (Fig. 6-6). Type the word or phrase (up to 24 characters) you want to find and press the enter key. The Model 100 searches the document from the cursor position and moves the cursor to the first occurrence.

To find the next occurrence of the same string, you just press F1 and the enter key. You do not have to retype the string. The Model 100 "remembers" the string until you type a new one or exit the document.

If the string you typed is not found, the Model 100 displays the prompt No match.

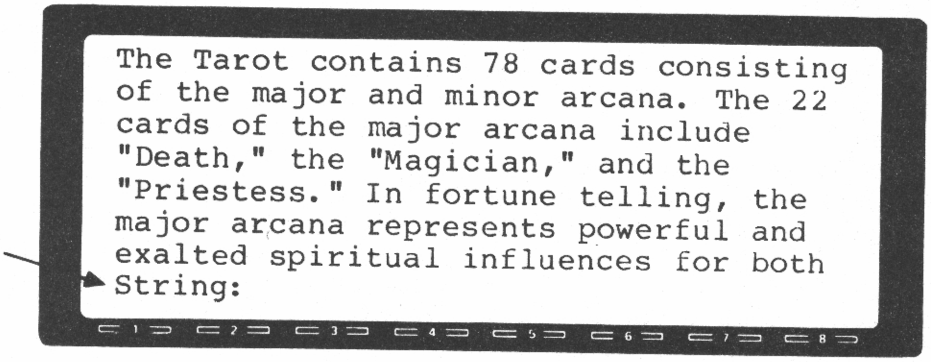
For example, you want to find the text that describes Malmar in our fantasy epic. You could use the shift and down arrow keys to scroll through the document a screenful at a time. Using the find key is quicker.

Hold down the control key and press the up arrow cursor key to move the cursor to the beginning of the document. Then press F1. The Model 100 prompts with String:. Type Malmar and press the enter key. The Model 100 moves the cursor to the first occurrence of *Malmar*.

To find the next occurrence of *Malmar*, press F1 and the enter key. The Model 100 moves the cursor to the next occurrence.

PRINTING TEXT

Yes, the Model 100 enables you to printout either just what's displayed on the screen as well as



```
The Tarot contains 78 cards consisting
of the major and minor arcana. The 22
cards of the major arcana include
"Death," the "Magician," and the
"Priestess." In fortune telling, the
major arcana represents powerful and
exalted spiritual influences for both
String:
```

Fig. 6-6. When you press F1, the Model 100 prompts you with the word *string*: for the word or phrase you want to find.

the entire document file. Yes, before you print, the Model 100 enables you to set the width of the printout. But that's it. The Model 100 offers none of the print features one would expect in a word processor. But let me say it again: you don't buy a portable for state-of-the-art word processing.

In order to print directly from the Model 100, you must, of course, connect it to a printer. The Model 100's printer interface uses the Model 100 Parallel Printer Cable to interface with most parallel printers and to all Radio Shack parallel printers.

You can also print through the RS-232 interface.

You can use the TELCOM program to upload a document from the Model 100 to a printer (See Part 5), and you can use the BASIC program's open command to output a document through the RS-232 to the printer (See Part 4).

The following print methods assume that you have connected a parallel printer to the Model 100's printer interface.

Printing the Screen

To print the contents of the screen, simply press the print command key. The Model 100 prints the eight lines of text appearing on the screen.

Printing a File

To print a document, first display it. Hold down the shift key and press the print key. The Model 100 prompts you for width with **Width:80**.

Press the enter key to keep the default (80), or type a different width and press the enter key.

The width determines the number of characters that will be printed on each line. You can set any width from 10 to 132 characters. For example, in 10 pitch, a printer prints 10 characters per inch, so a width of 80 characters gives you a printout 8 inches wide.

Chapter 7

Managing TEXT Documents

The Model 100 stores your TEXT documents in RAM and displays them in the menu. From time to time you will want to copy a document, delete one, or rename one. To *copy* a document you use the TEXT block commands select, copy, and paste. To *delete* or *rename* a document you use the BASIC commands kill and name.

COPYING THE DOCUMENT

If you are familiar with word processing you know that before you revise a document, you should make a copy (often called backup). Then, if you make a mistake, you still have your original. This is especially true on the Model 100. When you display and make changes in a document, the Model 100 automatically makes the changes in the RAM file. So, if you are not absolutely sure about the changes you want to make in a document, you should make a copy of the document before you revise it.

On the Model 100, you cannot directly copy a file. To make a copy of a document, you use the TEXT block commands.

Here is the idea. To copy a document, you display it, select the entire document as a block, and copy it. You then open a new document file by selecting TEXT and answering the displayed question **File to edit**. Press the paste key, and the model 100 will insert the copied document into the new document.

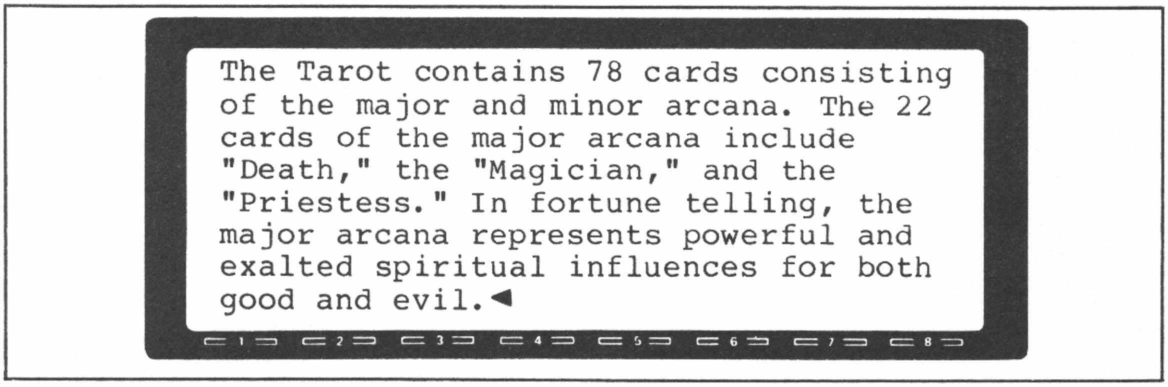
For example, to make a copy of a document named CARD.DO here are the steps you would use.

1. Select CARD.DO from the menu.

The document will be displayed (Fig.7-1).

2. Make sure the cursor is positioned at the beginning of the document and press the F7 key for Select.

3. Select the entire document to be copied by holding down the control key while pressing the



```
The Tarot contains 78 cards consisting
of the major and minor arcana. The 22
cards of the major arcana include
"Death," the "Magician," and the
"Priestess." In fortune telling, the
major arcana represents powerful and
exalted spiritual influences for both
good and evil.◄
```

Fig. 7-1. To copy a document, first display it on the screen.

down arrow cursor control key. The cursor moves down line by line to the end of the document (Fig. 7-2).

4. Press F5 to copy the document.

The Model 100 stores the document in the paste buffer.

5. Press F8 to exit the CARD.DO document. Since you only copied it, the CARD.DO file remains intact.

6. To open a new document return to the menu. Select TEXT from the menu, and answer the prompt **File to edit** with a name such as CRDSAF for Card Safe.

7. Press the paste key.

The Model 100 retrieves the CARD.DO document from the paste buffer and pastes or adds it to the new document.

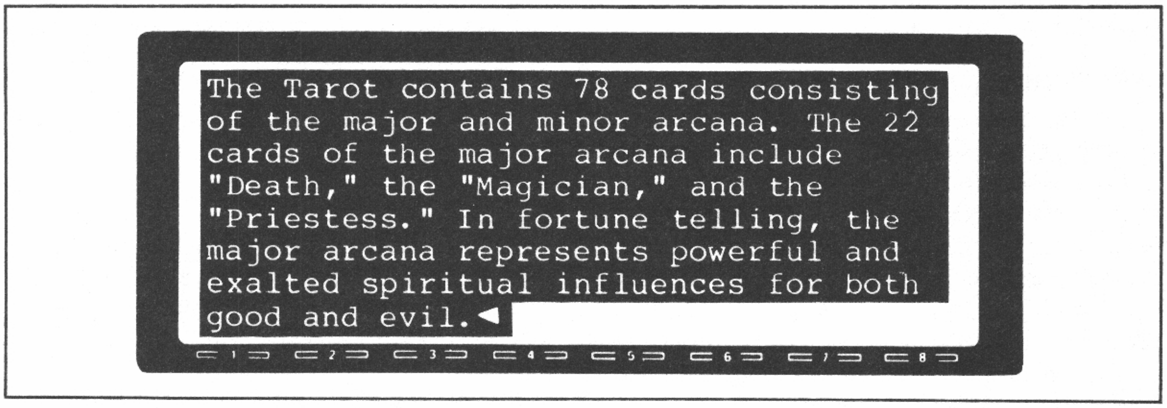
8. Press F8 to exit the new document.

You now have two identical documents, CARD.DO and CRDSAF.DO.

You can also write a BASIC program to copy TEXT documents, as explained in Part 4.

DELETING THE DOCUMENT

Documents take up space in RAM memory. Docu-



```
The Tarot contains 78 cards consisting
of the major and minor arcana. The 22
cards of the major arcana include
"Death," the "Magician," and the
"Priestess." In fortune telling, the
major arcana represents powerful and
exalted spiritual influences for both
good and evil.◄
```

Fig. 7-2. To select the entire document as a block, press the F7 key and hold down the control key and the down arrow cursor key until you reach the end of the document.

ment names take up space in the menu. If, for example, you continued to type documents, you could run out of space in the menu or in memory. So, when you are finished with a document, you should delete it to free up memory space. To delete a document, you use the BASIC kill command.

Here are the steps you would use to delete the CARD.DO document:

1. From the menu, position the cursor on BASIC and press the enter key.

The Model 100 displays:

```
TRS-80 Model 100 Software
Copr. 1983 Microsoft
##### Bytes free
Ok
```

2. Press F1 to display the list of files from the menu.

3. Type the BASIC kill command as given below.

```
KILL "CARD.DO"
```

You must type the name in quotes, and you must type the complete name, including the .DO extension.

4. Press the enter key.

BASIC kills or deletes the document.

5. Press F1 again to redisplay the RAM files and confirm that the document has been deleted (Fig. 7-4).

RENAMING THE DOCUMENT

Sometimes you may want to change the name of a document. To rename a document, you use the BASIC name command.

Here are the steps you would use to change the name of a document from CRDSAF.DO to CARD.DO.

1. From the menu, position the cursor on BASIC and press the enter key.

The Model 100 displays:

```
TRS-80 Model 100 Software
Copr. 1983 Microsoft
##### Bytes free
Ok
```

2. Press F1 to display the list of files from the menu.

3. Type the BASIC command as follows:

```
NAME "CRDSAF.DO" AS "CARD.DO"
```

You must type each name in quotes and the complete names, including the extensions.

4. Press the enter key.

BASIC renames the document.

5. Press F1 to redisplay the files and confirm that BASIC renamed your document (Fig. 7-5).

USING TEXT WITH CASSETTE

In TEXT, the F2 and F3 function keys enable you to save and load documents from RAM to cassette. Obviously, you must first connect the Model 100 to the cassette recorder. To do so, use the Model 100's built-in Cassette Interface with Radio Shack's Cassette Cable.

To use either load or save, you must first open a document file and display the document. Radio Shack's users manual sufficiently describes this feature, but here is a summary of how save and load work:

You press F2 to *load* a document from cassette to the Model 100. When the Model 100 prompts for a file name, type the name of the file you want and press the enter key.

The Model 100 then searches the tape. It displays the names of any files it skips over as it searches. When it finds the requested file, the

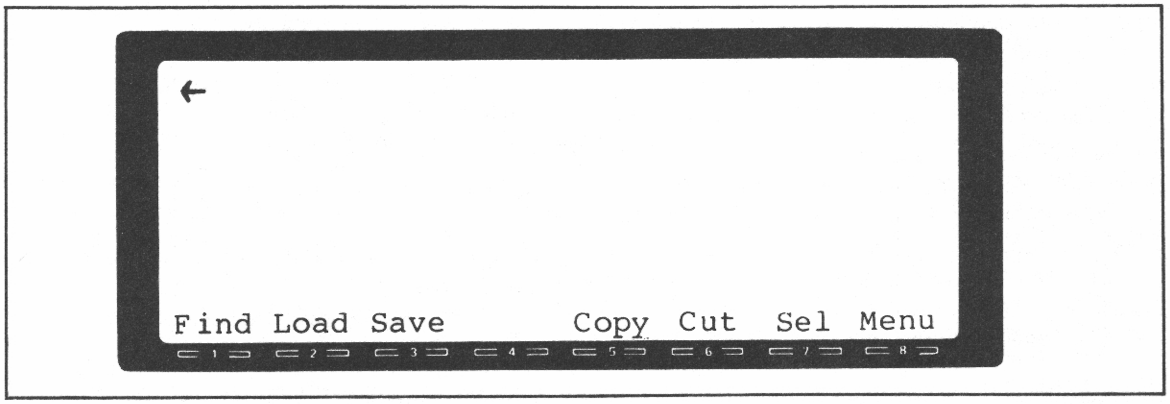
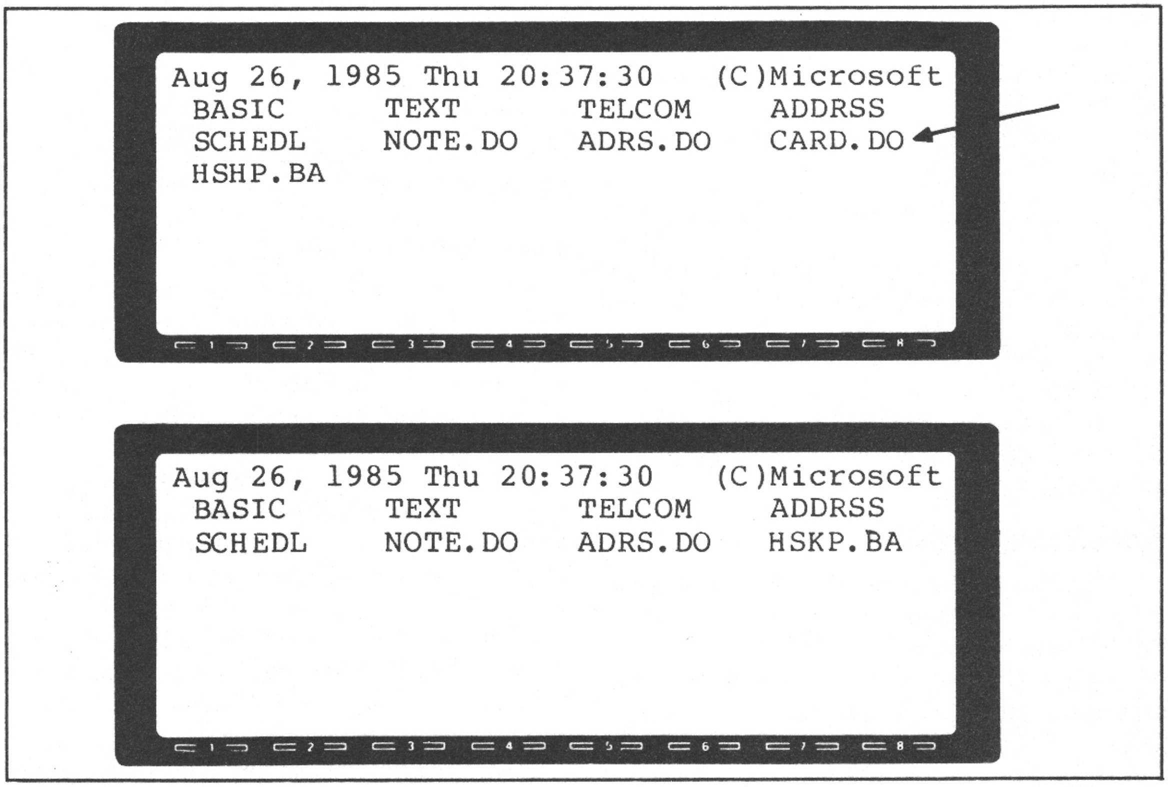


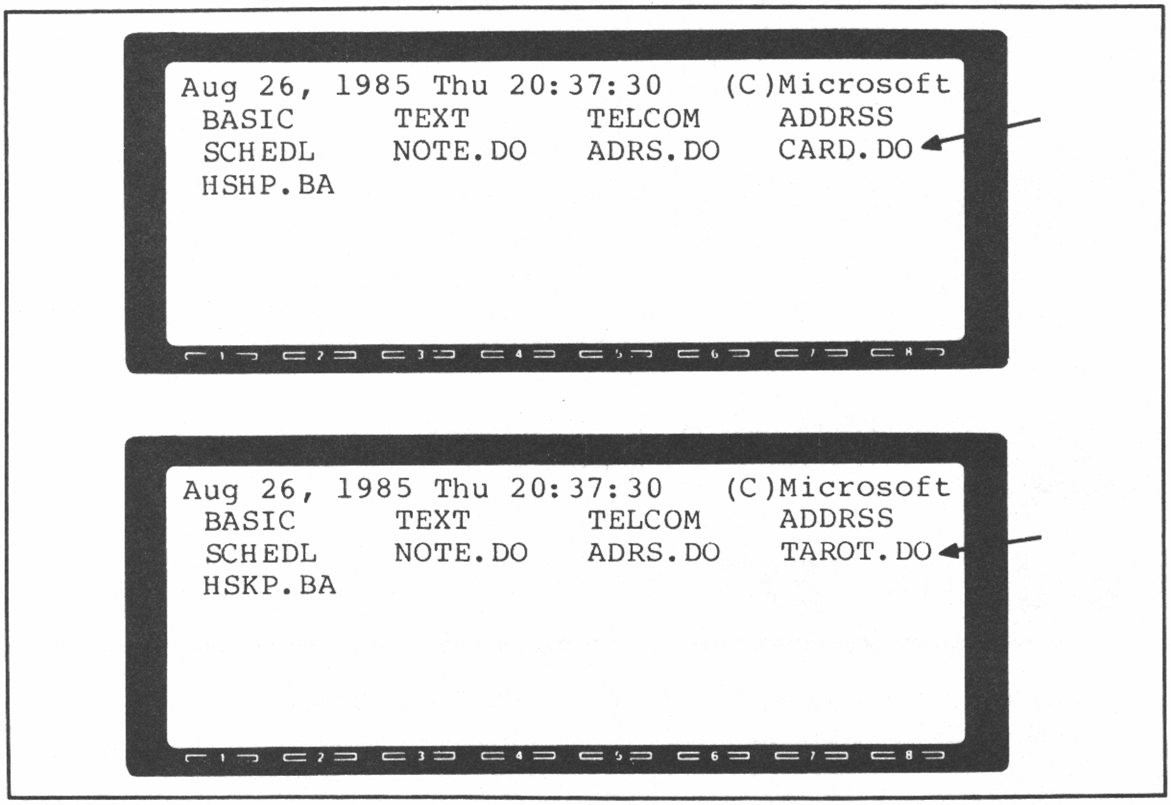
Fig. 7-3. After pressing F5 to copy the document into the paste buffer, return to the menu and select TEXT to open a new document.



```
Aug 26, 1985 Thu 20:37:30   (C)Microsoft
 BASIC      TEXT       TELCOM     ADDRSS
 SCHEDL     NOTE.DO    ADRS.DO    CARD.DO
 HSHP.BA
```

```
Aug 26, 1985 Thu 20:37:30   (C)Microsoft
 BASIC      TEXT       TELCOM     ADDRSS
 SCHEDL     NOTE.DO    ADRS.DO    HSKP.BA
```

Fig. 7-4. When you kill or delete a document from memory, the Model 100 automatically deletes the document name from the menu.



```
Aug 26, 1985 Thu 20:37:30   (C)Microsoft
 BASIC       TEXT       TELCOM     ADDRSS
 SCHEDL      NOTE.DO    ADRS.DO    CARD.DO
 HSHP.BA
```

```
Aug 26, 1985 Thu 20:37:30   (C)Microsoft
 BASIC       TEXT       TELCOM     ADDRSS
 SCHEDL      NOTE.DO    ADRS.DO    TAROT.DO
 HSKP.BA
```

Fig. 7-5. If you change a document name (CARD.DO to TAROT.DO for example), the new name will automatically be added to the menu.

Model 100 adds or appends the text from the cassette to the end of the document on display.

You press F3 to save a document to cassette from the Model 100. When the Model 100 prompts for a file name, type a name for the file and press the enter key.

The Model 100 then stores the document on cassette.

Chapter 8

Hands-On TEXT

Let's try out TEXT. This exercise will enable you to try out all of TEXT's features. Have fun!

If you have a printer and cable, hook up the printer. It is not absolutely necessary, however, to use a printer for this exercise.

Turn on the Model 100 and display the menu.

TYPING THE DOCUMENT

1. Display the menu.
2. Select TEXT.
3. When you see the prompt, File to edit?, name the document Cards.
4. Type the text in Fig. 8-1.

Use the delete/backspace key to correct any typographical errors as you type.

Use the tab key to position the headings as shown.

Type CODE q to type á as shown.

Use the graphics key with 2, 3, 4, and 5 to type ♧, ♡, ♤, and ◇.

COPYING THE DOCUMENT

It's always a good idea to make a copy of a document before you revise it. After you have typed the document in Fig. 8-1, you can copy it.

1. Press the label key to display the list of function key functions. Select the document as a block and copy it as follows.
2. Hold down the control key and press the up arrow cursor key to position the cursor at the beginning of the document.
3. Press F7 for Select.
4. Hold down the control key and press the down arrow cursor key to move the cursor to the end of the document to select the entire document as a block.
5. Press F5 to copy the block, in this case the entire document.
6. Press F8 to return to the menu.

Practice Copy

Playing Cards
The deck of cards in use today first appeared in fourteenth-century Europe. They probably evolved from Chinese fortune-telling cards by way of the Tárot.

Explanation
Use CODE q to type á

The Tárot is a European adaptation of the Chinese cards, and the contemporary 52-card deck derived from the Tárot.

The Tárot contains 78 cards consisting of the major and minor arcana. The 22 cards of the major arcana include Death, the Magician, and the Priestess. In fortune telling, the major arcana represents powerful and exalted spiritual influences for both good and evil.

Our present-day playing cards derive from the Tárot's minor arcana. The minor arcana consists of four suits, each containing 14 cards: The four suits are Wands, Cups, Swords, and Pentacles (a five-pointed star). These four suits come down to us as Clubs, Hearts, Spades, and Diamonds.

In the Tárot, each of the four suits represents a type of person or a spiritual proclivity.

TAB once to type Wands.
GRPH 2 to type ♣

    Wands ♣
This suit represents the fair, blonde and blue-eyed. In spiritual terms, Wands are the suit nearest to the major arcana and represents the highest aspirations of earthly existence.

Fig. 8-1. This typing exercise (continued on page 52) will get you using all of TEXT's features.

Use GRPH 4 to type ♡	Cups ♡ This suit represents the medium fair complexion: brown hair, blue or brown eyes. In spiritual terms, cups represent the aesthetic aspects of life, comfort, and gracious living.
Use GRPH 5 to type ♤	Swords ♤ The suit that stands for a swarthy complexion. (The famous tall dark stranger is the King or Prince of Swords). Spiritually, swords are interpreted as passions such as desire and anger.
Use GRPH 3 to type ♢	Pentacles ♢ This suit represents the dark complexion. Pentacles are concerned with the material aspects of living such as money and possessions.

Fig. 8-1. Continued from page 51.

Paste the copy into a new document as follows:

7. Select TEXT from the menu.
8. Answer the prompt **File to edit?** to name the new document. (How about using the name **CRDSAF**?) Press the enter key.
9. When the blank screen appears, press the paste key.

The Model 100 inserts the copied block into the new document.

10. Press F8 to return to the menu.

The menu will now display two documents, **CARD.DO** and **CRDSAF.DO**

REVISING THE DOCUMENT

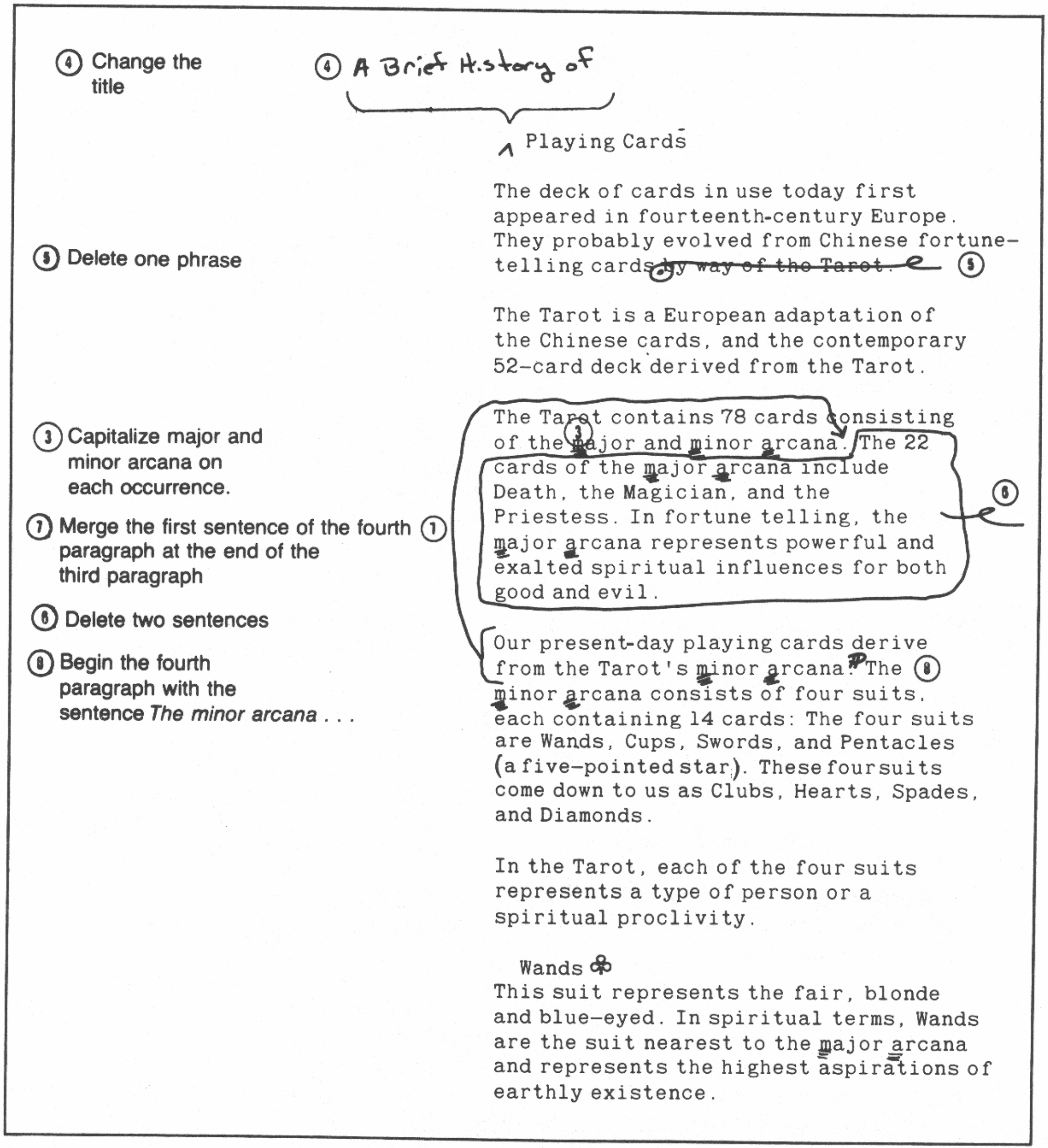
Now that you have a copy of your CARD document (CRDSAF), revise it. If you make a mistake, you still have the original intact. Revisions you might want to make are shown on Fig. 8-2. The procedure is as follows:

1. Use the label key to display the function key functions.
2. Hold down the control key and press the up arrow cursor key to make sure the cursor is positioned at the beginning of the document.
3. Use F1 (for Find) to change (capitalize) every occurrence of *major arcana* and *minor arcana* to *Major Arcana* and *Minor Arcana*.

☐ Find the first occurrence of *major arcana* and *minor arcana*.

☐ Press F1 for Find.

☐ Answer the **String:** prompt by typing **arcana**. Press the enter key.



④ Change the title

④ A Brief History of

Playing Cards

The deck of cards in use today first appeared in fourteenth-century Europe. They probably evolved from Chinese fortune-telling cards ~~by way of the Tarot~~. ⑤

⑤ Delete one phrase

The Tarot is a European adaptation of the Chinese cards, and the contemporary 52-card deck derived from the Tarot.

③ Capitalize major and minor arcana on each occurrence.

⑦ Merge the first sentence of the fourth paragraph at the end of the third paragraph ⑦

⑥ Delete two sentences

⑧ Begin the fourth paragraph with the sentence *The minor arcana . . .*

The Tarot contains 78 cards consisting of the major and minor arcana. ③ The 22 cards of the major arcana include Death, the Magician, and the Priestess. In fortune telling, the major arcana represents powerful and exalted spiritual influences for both good and evil. ⑥

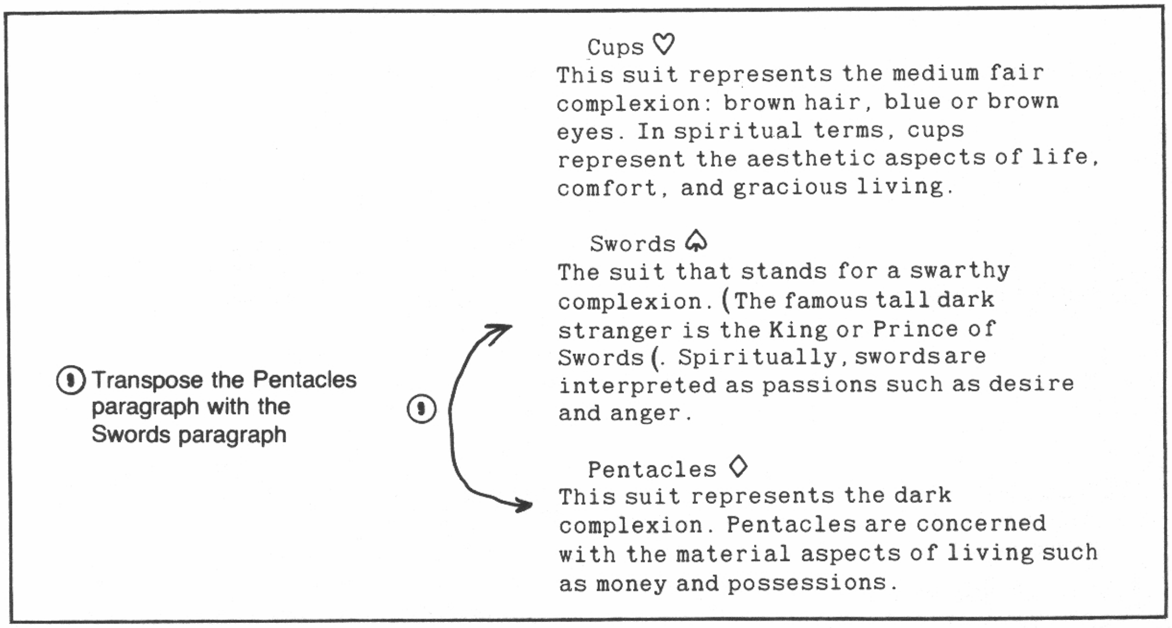
Our present-day playing cards derive from the Tarot's minor arcana. ¶The ⑧ minor arcana consists of four suits, each containing 14 cards: The four suits are Wands, Cups, Swords, and Pentacles (a five-pointed star). These four suits come down to us as Clubs, Hearts, Spades, and Diamonds.

In the Tarot, each of the four suits represents a type of person or a spiritual proclivity.

Wands ♣

This suit represents the fair, blonde and blue-eyed. In spiritual terms, Wands are the suit nearest to the major arcana and represents the highest aspirations of earthly existence.

Fig. 8-2. Change your document as shown here and on page 54 to practice making revisions. The circled numbers refer to the numbered steps in the text.



⑨ Transpose the Pentacles paragraph with the Swords paragraph

⑨

Cups ♡
This suit represents the medium fair complexion: brown hair, blue or brown eyes. In spiritual terms, cups represent the aesthetic aspects of life, comfort, and gracious living.

Swords ♤
The suit that stands for a swarthy complexion. (The famous tall dark stranger is the King or Prince of Swords (. Spiritually, swords are interpreted as passions such as desire and anger.

Pentacles ◇
This suit represents the dark complexion. Pentacles are concerned with the material aspects of living such as money and possessions.

Fig. 8-2. Continued from page 53.

The Model 100 will move the cursor to the first occurrence of *arcana*.

☐ Press CAPS LOCK.

☐ Use the shift key with the delete/backspace key to delete the letter *a*.

☐ Type A to insert a capital letter.

☐ Use the shift key with the left-pointing arrow cursor key to move the cursor back to the *m* in *major*.

☐ As you did with the a, delete the m and insert an M.

Find and change all other occurrences, just press F1 and the enter key. The Model 100 will retain the string until you type a new one or return to the menu.

☐ When there are no more occurrences, the Model 100 will display No match.

4. Use insert to change the title to A Brief History of Playing Cards.

☐ Use the control key and up arrow cursor key to move the cursor to the beginning of the document.

☐ Position the cursor on the *P* in *Playing*, and type the new title.

5. Use the shift and delete/backspace keys to delete *by way of the Tarot.* If you delete the period, insert another one.

6. Use F7 (select) and F6 (cut) to delete the two sentences beginning with *The 22 cards in the Major Arcana . . .*

☐ Position the cursor on the *T* in *The*.

☐ Press F7.

☐ Press the down arrow cursor key six times to select the rest of the paragraph as a block.

☐ Press F6 to cut (delete) the block.

7. Merge the next paragraph beginning with *Our present day . . .* into the preceding paragraph beginning with *The Tarot contains 78 . . .* by using the shift and delete/backspace keys to delete the "Returns" (◀).

8. Insert two returns to break the text into paragraphs as shown so that the fourth paragraph begins with the sentence *The Minor Arcana consists . . .*. To do so

☐ Delete the spaces after the period.

☐ Make sure the cursor is positioned on *T* in *The Minor*.

☐ Press the enter key twice to insert two returns.

9. Transpose the order of the Pentacle and Sword paragraphs by using the F7, F6, and paste keys to select, cut, and paste the document.

To select and cut the Pentacles paragraph:

☐ Position the cursor at the left margin of the line that contains the heading Pentacles ◇ .

☐ Press F7 for Select.

☐ Hold down the control key and press the down arrow cursor key to define the rest of the text as a block.

☐ Press F6 to cut the block.

To paste the Pentacles paragraph before the Sword paragraph:

☐ Position the cursor at the left margin of the line that contains the heading Swords ♤.

☐ Press the paste key.

FILE MANAGEMENT

Once you have revised your document, use BASIC's file management commands to delete the original, unrevised document from RAM and to rename the copy.

If you revised the original document rather than the copy, then you would delete the copy. This is a better way to do it, but then you would not have been able to try out the name command.

1. From the menu select BASIC.
2. Press the label key to display the function key functions.
3. Press F1 to display the files as they appear on the menu.
4. Delete (kill) the original document (CARD.DO in our example) from RAM.

Type the following command exactly as it appears including the quotation marks.

```
KILL"CARD.DO"
```

5. When BASIC displays Ok, press F1 again to verify that CARD.DO is gone.
6. Use BASIC's name command to rename CRDSAF as CARD.

```
NAME "CRDSAF.DO" AS "CARD.DO")
```

PRINTING THE DOCUMENT

Now that you have typed and revised the document, print it.

1. Make sure the Model 100 is correctly connected to the printer and that the printer is powered up and on line.
2. From the menu select CARD.DO.
3. Try printing the eight lines on the screen by pressing the print key.
4. Print the entire document and try out different widths.

☐ Hold down the shift key while you press the print key.

☐ Use the default width of 80 characters by pressing the enter key.

☐ Hold down the shift key and press the print key.

☐ Answer the width prompt by typing **40** then press the enter key.

☐ Hold down the shift key and press the print key.

☐ Answer the width prompt by typing **65**, then press the enter key.

PART 3
SCHEDL and ADDRSS

I cover the schedule program (SCHEDL) and the address program (ADDRSS) together because they are basically the same program. Both are "souped up" data base managers that enable you to find data quite rapidly. Once you know how to use one of these programs, you know how to use both of them. The only difference between the two is that ADDRSS has additional uses with TELCOM, which are covered in Part 5 on Communications.

In order to use SCHEDL and ADDRSS you must know how to use the TEXT program to open a document, type text, and revise text. If you do not know how to use the Model 100's TEXT program, read Part 2 before you continue with Part 3.

Chapter 9

Introduction

SCHEDL and ADDRSS are simple to use. First you use TEXT to create a document as a data base. Into each document, you type data such as name, address, and phone number. Once you have typed the data into the data base, you use SCHEDL and ADDRSS to quickly find an item or group of items. Using the programs is so easy it's ridiculous. The real art is setting up your data bases.

For example, to use ADDRSS, you must first use TEXT to create a document named ADRS.DO. In the document you type your records. Each record will probably include a name, address, and phone number.

Once you have created the ADRS.DO data base, the ADDRSS program enables you to locate any of the records in your data base in the twinkling of an eye. For example, when you select the ADDRSS program, and press F1 for find, you type Roy, say, to find the phone number of Roy Colins. ADDRSS searches the ADRS.DO data base, locates Roy, and displays the complete record on the screen.

If you know just a smidgen about data base management, you can use SCHEDL and ADDRSS more effectively.

DATA BASE MANAGEMENT

A *data base* is a collection of data. Usually, related items are grouped together as one record. The data base is a list of records. A phone book is a perfect example of a data base. In the phone book, each listing is one record containing last name, first name, address, and phone number.

In data bases, the order of data in each record is important. In each record you must list the data in the same sequence as given in every other record. In the phone book, each record starts with a last name, which is followed by a first name, an address, and finally a phone number. The first item in each record should be the one you will most often use to find the record. In the phone book, for example, you search records by last name, so the last name is listed first.

Data base management is the way computers use data bases to store records, sort them, and find them. To find a record in the phone book, you search alphabetically by last name. What if you wanted to search by phone number, address, or even first name? Solving a problem like this is one challenge that absorbs many programmers.

Data base management is a big deal. You can imagine how important a good data base management program is to a company with a lot of data, such as a credit card company or a bank. Programmers are always vying to come up with a better data base management program, one that stores the most data in the least space and that can find specific data most quickly. SCHEDL and ADDRSS are pretty fair, small-scale data base management programs.

THE DIFFERENCES BETWEEN SCHEDL AND ADDRSS

So far I have concentrated on the similarities between ADDRSS and SCHEDL. They are indeed similar, so why have two such similar programs?

Well, you should really use ADDRSS as an automated address book for names, addresses, and telephone numbers. With TELCOM or BASIC, you can use the data in your ADDRSS data base (ADRS.DO) to find and dial numbers. What's more, you can use the ADRS.DO data base with TEXT to paste an address block in a letter and to address an envelope. You really should reserve ADDRSS for use as a computerized address book.

SCHEDL is designed for keeping track of dates such as appointments, deadlines, expenses, and so on. Truth be told, you can use the super-fast SCHEDL program to keep track of *any* data that you need including accounts, orders, inventory, birthdays, your record collection . . . you name it.

The point is that since you must dedicate one data base management program to addresses (ADDRSS), it's good to have another one (SCHEDL) for other purpose's including, of course, schedules.

Creating and using the data bases for ADDRSS and SCHEDL are covered in Chapters 10 and 11, but that's just part of the story. Part 4 shows you how to use the ADDRSS data base (ADRS.DO) to dial numbers and to log on to commercial data bases such as CompuServe. Chapter 20 suggests some BASIC programs you can use to update your data bases.

Chapter 10

Creating Data Bases

Before you use SCHEDL and ADDRSS, you must first use TEXT to create a data base.

☐ For SCHEDL, you create a document named NOTE.DO.

☐ For ADDRSS you create a document named ADRS.DO.

SCHEDL and ADDRSS enable you to find records in your data base almost instantly. What really determines the effectiveness of these programs is how well you set up the data bases at the beginning. True to the expression, "garbage in, garbage out," ADDRSS and SCHEDL are only as good as the data bases you create for them.

PLANNING A DATA BASE

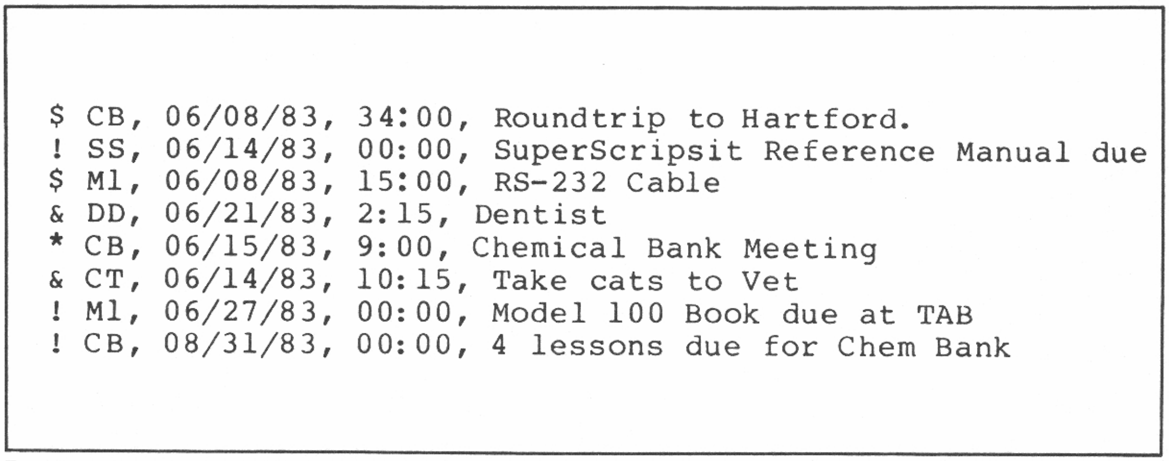
Before you create a data base, you should put some thought into it. Think of the most useful way to organize the data. To give you an idea of what I mean by planning a data base, an example follows.

When creating the NOTE.DO data base for SCHEDL, think about the *kind* of schedule data you want in each record and rank each kind of data in order of importance. Will you most often look for data on a project-by-project basis? A day-of-the-week basis? Or on a client basis?

Let's say you are a freelance writer with three projects called The Model 100 Book, The Super-Scripsit Reference Manual, and The Chemical Bank Project.

Each project has its own set of deadlines, appointments, and expenses. It may be most useful for you to order each record by project, followed by date, followed by description, followed by note. So you would plan your data base as shown in Fig. 10-1.

The idea of planning your data base will make more sense after you have tried it. So, let me show



```
$ CB, 06/08/83, 34:00, Roundtrip to Hartford.
! SS, 06/14/83, 00:00, SuperScripsit Reference Manual due
$ Ml, 06/08/83, 15:00, RS-232 Cable
& DD, 06/21/83, 2:15, Dentist
* CB, 06/15/83, 9:00, Chemical Bank Meeting
& CT, 06/14/83, 10:15, Take cats to Vet
! Ml, 06/27/83, 00:00, Model 100 Book due at TAB
! CB, 08/31/83, 00:00, 4 lessons due for Chem Bank
```

Fig. 10-1. A sample NOTE.DO data base.

you how to create the two data bases NOTE.DO (for SCHEDL) ad ADRS.DO (for ADDRSS).

CREATING NOTE.DO

Before you use SCHEDL, first use TEXT to create a document named NOTE.DO as your data base. Then use SCHEDL to search the NOTE.DO document to quickly find the records you want.

Here are the steps you use to create the NOTE.DO data base for SCHEDL:

1. Display the menu and select TEXT.
The Model 100 prompts:

File to edit?

2. Type NOTE and press the enter key.

The Model 100 displays the blank screen. You are ready to type your records.

3. Type each record. Press the enter key after each record is typed.

4. When you have typed all the records you want, press F8 to exit TEXT and return to the menu.

Your data base is now listed in the menu as NOTE.DO.

Pointers on Typing NOTE.DO Records

First of all, don't let the name SCHEDL hold you back. You can use SCHEDL to keep track of any data that is useful in list form.

As illustrated in Fig. 10-1, here are some pointers for typing your records in the NOTE.DO data base:

1. At the beginning of each record, type a symbol to identify the record. For example, type an exclamation point (!) for deadlines, an ampersand (&) for appointments, and a dollar sign ($) for expenses.

2. In each record, type first the item you will most often search for. For example, if you will most often search for projects, type the project name first in each record.

3. Separate each item in a record by typing a space.

4. End each record by pressing the enter key. TEXT displays ◄ .

CREATING ADRS.DO

As I said in Chapter 9, here is the main difference between SCHEDL and ADDRSS: while you can use SCHEDL to store and locate almost any kind of data, you should use ADDRSS for names, addresses, and phone numbers. Why? ADDRSS is not the only program that uses the ADRS.DO data base. TELCOM uses ADRS.DO to dial telephone numbers and to log on to information services such as CompuServe. You can also write BASIC programs that use the ADRS.DO data base to dial numbers and log on.

Because TELCOM uses the ADRS.DO data base to dial the telephone, you must enclose each phone number in colons. (The colons enable TELCOM to identify the phone number in the record.) For example, when you type the number (212) 569-0408 in the ADRS.DO data base, you must type it as

```
:2125690408:
```

Here are the steps you use to create the ADRS.DO data base for ADDRSS:

1. Display the menu and select TEXT.
The Model 100 prompts with

```
File to edit?
```

2. Type ADRS and press the enter key.
The Model 100 displays the blank screen. You are now ready to type your records.
3. Type each record and press the enter key after each record. Remember to enclose the telephone numbers in colons.
4. Press F8 to return to the menu.

POINTERS ON TYPING ADRS.DO RECORDS

As illustrated in Fig. 10-2, here are some pointers for typing your records in the ADRS.DO data base:

1. At the beginning of each record, type a symbol to classify the record. For example, type `!F` for family, `!B` for business, `!P` for personal.
2. Enclose each phone number in colons, such as `:2125690408:` .
3. Type first the items you will most often search for. For example, if you will most often search for last names, type the last name first in each record.
4. Separate each item in a record by typing a space.
5. End each record by pressing the enter key. TEXT will display ◄ .
6. When you have typed all the records you want, press F8 to exit TEXT and return to the menu.

Your data base will be listed in the menu as ADRS.DO.

UPDATING DATA BASES

The fact that you use TEXT to create and

```
!B Stuart Gibson, Concept Industries, 1116 Summer Street,
Stamford, Connecticut 06905 :5550982:
!F Mom and Dad, 300 Home Street, Greensboro, NC 27400
:9192997007:
!P Mimi Larue, 618 21st Street, New York, NY 10014 :6829000:
```

Fig. 10-2. A sample ADRS.DO data base for use with the ADDRSS program.

Open the NOTE.DO document.

Press the control key and the down arrow cursor key to move to the end of the document.

Type the following record:

```
!M, 07/04/85, 10:00, Dentist, Phil Ling
```

Press the enter key.

Use the F7 key (Select) and the F5 key (Copy) to copy the record.

Press the paste key four times to insert the record, which will appear as:

```
!M, 07/04/85, 10:00, Dentist, Phil Ling
!M, 07/04/85, 10:00, Dentist, Phil Ling
!M, 07/04/85, 10:00, Dentist, Phil Ling
!M, 07/04/85, 10:00, Dentist, Phil Ling
!M, 07/04/85, 10:00, Dentist, Phil Ling
```

Finally, use insert and the delete/backspace key to edit each record to reflect the dates and times for each specific appointment. The revised document will look something like:

```
!M, 07/04/85, 10:00, Dentist, Phil Ling
!M, 07/15/85, 12:00, Dentist, Phil Ling
!M, 07/21/85,  2:00, Dentist, Phil Ling
!M, 08/05/85, 10:00, Dentist, Phil Ling
!M, 08/11/85,  2:00, Dentist, Phil Ling
```

Fig. 10-3. The procedure for updating a data base using the TEXT program.

revise the two data bases is no accident. You will often want to update your data bases, and nothing could be easier than using TEXT to revise a data base.

For example, on August 8, you will probably want to delete the appointments for August 7 from NOTE.DO. You will also want to add new appointments to NOTE.DO as you make them. The Model 100's TEXT program makes it easy to insert, delete, and revise records for NOTE.DO and ADRS.DO. To update data bases

1. Display the menu, and position the cursor on the data base (either NOTE.DO or ADRS.DO) you want to update.
2. The Model 100 will display the document.

Use the TEXT commands to revise the document.

3. Press F8 to return to the menu.

TAKING ADVANTAGE OF TEXT

For both data bases, ADRS.DO and NOTE.DO, be sure to take full advantage of the TEXT program for typing and updating your records. Here are two ideas for keeping your data bases up to date.

Let's say you must add several similar records—five appointments with your dentist, Dr. Phil Ling—to your NOTE.DO data base. Use the procedure in Fig. 10-3.

When you no longer need a record, be sure to delete it. Use F7 (Select) to select the record. Use F6 (Cut) to delete it. Of course, you can also use the delete/backspace key to delete a record one character at a time.

A BASIC REMINDER

You can write BASIC programs to update and sort your data bases. For example, you may want to sort the NOTE.DO data base by date, or you may want to sort your ADRS.DO data base alphabetically by last name. Chapter 20 contains programs that illustrate some ways to manage your NOTE.DO and ADRS.DO data bases.

Chapter 11

Using SCHEDL and ADDRSS

As I discussed in Chapter 10, creating the data base is the real art. Once you have done that, using either SCHEDL or ADDRSS to find items is a piece of cake.

The following steps illustrate how to use SCHEDL and ADDRSS after you have created the appropriate data base.

1. Display the menu.
2. Position the cursor on either **SCHEDL** or **ADDRSS** (Fig. 11-1) and press the enter key.

☐ If you choose ADDRSS, the Model 100 displays **Adrs:** at the top of the screen.

☐ If you choose SCHEDL, the Model 100 displays **Schd:** at the top of the screen.

The Model 100 automatically displays the available function keys for SCHEDL or ADDRSS (Fig. 11-2). Press the label key to clear the list of available function key functions from the display.

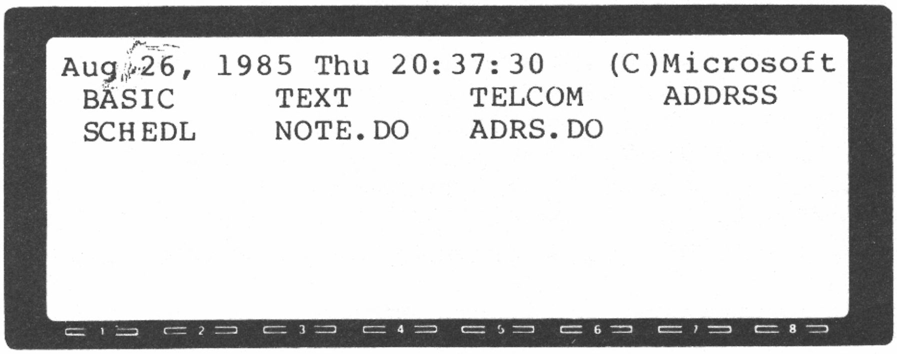
3. Press F1 or F5 to find one or more records.

☐ F1 enables you to find and list records on the *display*. If you press F1, the Model 100 displays **Find**.

☐ F5 enables you to find and *print* records. If you press F5, the Model 100 displays **Lfind**.

4. Type part of the record you want to find and press the enter key. To list all of the records, just press the enter key, no typing is necessary.

☐ To find one or more *specific* records, type part of the record and press the enter key.



```
Aug 26, 1985 Thu 20:37:30   (C)Microsoft
 BASIC      TEXT       TELCOM     ADDRSS
 SCHEDL     NOTE.DO    ADRS.DO
```

Fig. 11-1. To choose either the address or schedule program from the menu, position the cursor over either ADDRSS or SCHEDL and press the enter key.

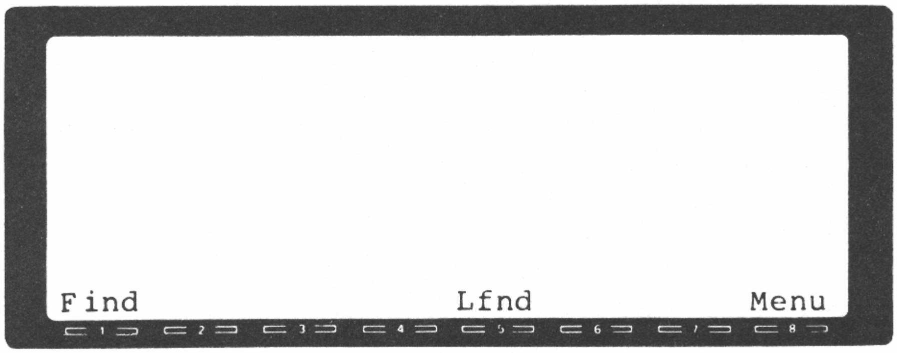
For example, if you are using ADDRSS and the ADRS.DO document contains several telephone numbers for Stuart Gibson, type Stuart and press the enter key. The Model 100 displays or prints all the records that contain the letters Stuart.

Or, if you are using SCHEDL and your NOTE.DO document contains several appointments for July, type 07/04 and press the enter key. The Model 100 displays or prints all the records that contain the characters 07/04.

☐ To display or print *all* the records in your data base, just press the enter key.

The Model 100 displays one screenful, or eight lines, of records at a time. If the Model 100 finds more than eight records, it displays the following message in line 8:

```
More Quit
 3    4
```



```
Find                Lfnd              Menu
```

Fig. 11-2. The function key definitions for ADDRSS and SCHEDL.

```
$ CB, 06/08/83, 34.00, Roundtrip to Hartford.
! SS, 06/14/83, 00:00, SuperScripsit Reference Manual due
$ M1, 06/08/83, 15.00, RS-232 Cabel
& DD, 06/21/83, 2:15, Dentist
* CB, 06/15/83, 9:00, Chemical Bank Meeting
& CT, 06/14/83, 10:15, Take cats to Vet
! M1, 06/27/83, 00:00, Model 100 Book due at TAB
! CB, 08/31/83, 00:00, 4 lessons due for Chem Bank
```

Fig. 11-3. A sample NOTE.DO data base.

5. Press F3 or F4.

☐ Press F3 (More) to display the next screenful.

☐ Press F4 (Quit) to quit displaying records.

6. Continue to display records by pressing F1 or F5 or press F8 to return to the menu.

For example, let's say your NOTE.DO data base looks like the one shown in Fig. 11-3. To display all of the records on the Model 100 (M1) project, you follow these steps.

1. Select SCHEDL from the menu.
2. Press F1 for Find.
3. Type **M1** to find all the Model 100 records and press the enter key.

The Model 100 will display all of the records for the Model 100 project.

PART 4
TELCOM

TELCOM stands for "telecommunications." The TELCOM program is the built-in software that enables the Model 100 to dial telephone numbers and to communicate with other computers and information services over the telephone or RS-232.

The Model 100's superb communications ability makes it a dynamite portable. You can easily transmit documents or programs written on the Model 100 to another computer.

If computer communications is new to you, let's face it, it is a little tricky to get started. Chapter 12 explains communication fundamentals and then briefly surveys the Model 100's TELCOM program. If you are already familiar with telecommunications and TELCOM, you may want to skim through the chapter and then skip ahead to Chapters 13 and 14, which get down to business. Chapter 13 shows you how to use the Model 100's modem. Chapter 14 shows you how to use the Model 100's RS-232 interface.

Chapter 12

Introduction

Let's communicate about communications on the Model 100. *Communication* means two or more computers sending and receiving data between each other. Have you ever cashed a check in an automated bank? The teller's computer terminal asks the bank's main computer if your account can cover the check. The computer terminal and the computer communicate.

Communications is a really big deal. There is a lot of information out there. Armed with a computer, a modem, and the right software, it's all yours. What's more, you can "talk back." Everything you need for first-rate communications is standard equipment on the Model 100. It may be the world's smartest terminal. What's more, the Model 100 comes equipped with everything you need to communicate directly over an RS-232 cable with your desk-top computer.

The Model 100 is a top-notch communicator. If you own an Apple or TRS-80 Model III, for example, you must buy additional hardware (a modem) and additional software (a communications software package) to communicate with another computer. On the Model 100, all this is standard equipment. The Model 100 contains a built-in modem, an RS-232 interface, and a communications program—TELCOM. The minute you get the computer, you are ready to go "on line" with the world.

For example once you hook the Model 100's modem to your telephone, the world is your oyster. There are well over a thousand data bases including stock market quotes, recipes, encyclopedias, free software, games, available to you. The list is endless. The number of data bases is growing. Some of the most famous data bases are CompuServe, Dow Jones, and The Source. There are also clubs and local groups, called bulletin boards, you can "tie-in" with. You can shop, do your banking, get the latest baseball scores, check out car prices and more from such data base sources. You can "talk" to other users, leave messages for other computerists, and research arcane subjects such as Malaga pottery through computer-to-computer hookups.

It's an embarrassment of riches, and it is precisely this richness, combined with a few technical requirements that create "communications phobia" in many novices. Let's face it; if you have come this far, you are ready for communications.

A CRASH COURSE IN COMMUNICATIONS

There is nothing really mysterious about communication. In effect, communication is going on all the time *within* a computer. The keyboard sends data to the memory and screen, the memory sends data to the printer, and so on.

Let's backtrack a moment. Computers work with *bits*, 1's and 0's. Now a single bit doesn't mean very much, but put eight (eight 1's or eight 0's or a combination of 1's and 0's) bits together, and you can represent numbers from 0 (00000000) to 255 (11111111) in the *binary*, or base 2, number system. Eight bits make up what's called *byte*. The computer stores all numbers in binary form.

Okay, what about letters of the alphabet? When you use TEXT to type **Dear Ray**, how does the Model 100 store that in RAM? Each character has a unique code (an 8-bit number) that is universally recognized. This set of codes is called ASCII (pronounced "ass-key"), an acronym for American Standard Code for Information Interchange. In ASCII, *A* is 65 (01000001), *B* is 66 (01000010), *C* is 67; a lowercase letter *a* is 97, *b* is 98, *c* is 99. A space is 32, and an exclamation point (!) is 33. A table of ASCII code is given in the Appendix.

If you type **Dear Ray** (8 characters), the Model 100 stores eight 8-bit ASCII codes in RAM: 68, 101, 97, 114, 32 (space), 82, 97, and 121. The software that controls the display knows ASCII, so when you type **D**, the central processing unit in the computer sends a 68 (01000100) to the display control, which displays a **D**.

Do you see what this has to do with communications? When you send data from one computer to another, the computer sends a stream of 10-bit "chunks" (Fig. 12-1). The first bit says, "here comes an 8-bit ASCII character," the next 8 bits are the character, and the tenth bit says "that's the end of this character." The first bit is called, logically enough, a *start bit* and the tenth bit is called a *stop bit*. So, when computers send data, they send *bit streams*, a series of 1's and 0's.

For example, let's say you used TEXT to type a letter on the Model 100. You named the document SALES. Now you want to upload or transfer the document from the Model 100 to your Model III and use a word processing program to revise it and print it.

You use an RS-232 cable to connect the Model

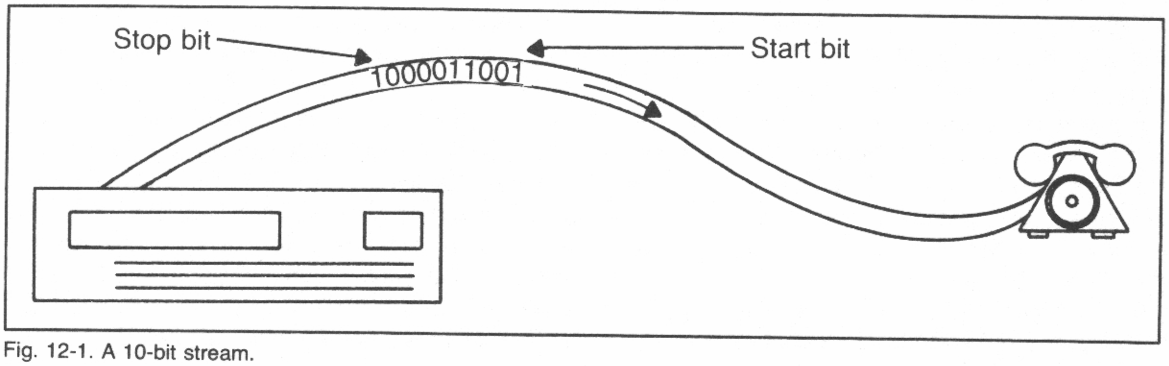


Fig. 12-1. A 10-bit stream.

100 to your Model III, then you upload or transfer the SALES document from the Model 100 to the Model III. The Model 100 opens the SALES document, reads each letter (as an 8-bit ASCII number), and sends the bits to the Model III. The Model III receives the bits and stores them on disk. You have "uploaded" SALES from the Model 100 to the Model III.

You can also use the modem to connect the Model 100 to the telephone and "log on" (connect) with the CompuServe information service. You could then ask CompuServe for information such as car prices. CompuServe will send the ASCII codes over the phone, and the Model 100 will display it as prices.

CONNECTING COMPUTERS

So much for a vastly simplified description of the fundamentals. Let's get specific. To communicate, there are two primary requirements.

- ☐ The computers must be "connected," obviously.
- ☐ The computers must speak the same "language."

There are two main ways to connect computers: by RS-232 interface or by *modem*.

RS-232 Communications

RS-232 is the name of a standardized method (interface) and hardware (cable) for transmitting data. Any RS-232-equipped computer can communicate with any other RS-232-equipped computer. Most computers include RS-232 interfaces as standard equipment.

There is nothing mysterious about RS-232 either. It's just a convention. The Electronic Industries Association (EIA) got together and agreed upon a universal method for sending bits. So whether you own an Apple or a Xerox computer, it's possible to transmit data via an RS-232.

EIA said in effect, "let's send data serially and asynchronously." In simple terms that means, "let's send data one bit at a time, using start and stop bits to separate units.

Modem Communications

Although you can use a modem to connect two computers in the same room, usually a modem is the device you use to communicate over the telephone.

Modem stands for "modulator/demodulator." Conceptually, it's not that different from FM (frequency modulation) radio. A modem is a device that converts the computer's signals or bits into sound frequency, a form that can be transmitted over the telephone.

On the sending end, the modem modulates the bits into sound frequencies that can travel over a telephone line. On the receiving end, the modem demodulates the frequencies and converts them back into bits.

Speaking the Same Language

Whether you transmit bits over a modem or RS-232, both computers must speak the same language; they must have identical communications parameters. The Model 100 calls this STAT for communications "status."

RS-232 convention consists of five parameters. Within each parameter you have a range of choices. For example, *baud* is the speed of the transmission. At 300 baud, your computer transmits 300 bits per second (thirty 10-bit units). If the other computer is receiving at 4,800 baud (480 words per second), you will have a failure to communicate.

At the start, therefore, you must make sure that both computers are set to the same parameters. These parameters are for

Baud
Word Length
Parity
Stop Bit
Line Status
Pulse Rate (for automatic dialing)

Baud. The term *baud* comes from J. M. E. Baudot, a nineteenth century French inventor who improved on Morse Code transmission. Generally, the term baud has nothing to do with his invention. It has come to stand for "bits per second."

When you communicate, both computers must be set at the same baud. If you are sending at 300 bits per second and the other computer is receiving at 9,600 bits per second, you will end up with gibberish. You can set the Model 100 to transmit from 300 to 19,200 baud.

Word Length. Computers transmit bits in groups called words. A *word* is either 6, 7, or 8 bits at a time and does not include the stop bit or start bit. Both computers must be set to communicate at the same word length.

Parity. Parity is a method for double checking the accuracy of the communication. For example, if the computer is supposed to send 6-bit words, the parity should be "even." If it is sending 7 bits, the parity should be "odd." The Model 100 enables you to ignore parity, to use odd parity, to use even parity, or to use no parity at all.

Stop Bit. This is the number of bits used to signal the end of a word. It is either 1 or 2. Again, you must make sure that both computers are set to recognize the same number of stop bits.

Line Status. The line status setting tells the computer whether to wait for a ready signal from the other computer. You usually enable (activate) the wait if one computer is slower than the other.

On the Model 100 you can set the line status as E for enable (enable the wait feature) or D for disable (no wait).

SOME OTHER TERMS YOU SHOULD KNOW

Modem, RS-232, ASCII, and communications parameters are the major terms you need to know, but should also know a few others. These are

Upload and Download
Full Duplex and Half Duplex.
Originate and Answer

Upload and Download

It all depends on your point of view. When you send your document from the Model 100 to your Model III, the Model 100 uploads—sends out—the file (Fig. 12-2). The Model III downloads—receives and stores—the document (Fig. 12-3).

Both upload and download refer to files that are stored somewhere, either in RAM, on cassette, or on a disk.

Both upload and download refer to files that are stored somewhere, either in RAM, on cassette, or on a disk.

☐ *Upload* means to *send* a file from your computer to another.

☐ *Download* means to *receive* data and store it as a file.

Duplex (Full and Half)

No, it's not a two-apartment unit. The term *duplex* means that both computers send and receive at the same time. In *full duplex*, the data you send goes straight to the host computer, and the host's data goes straight to you.

In *half duplex*, Computer A sends, and Com-

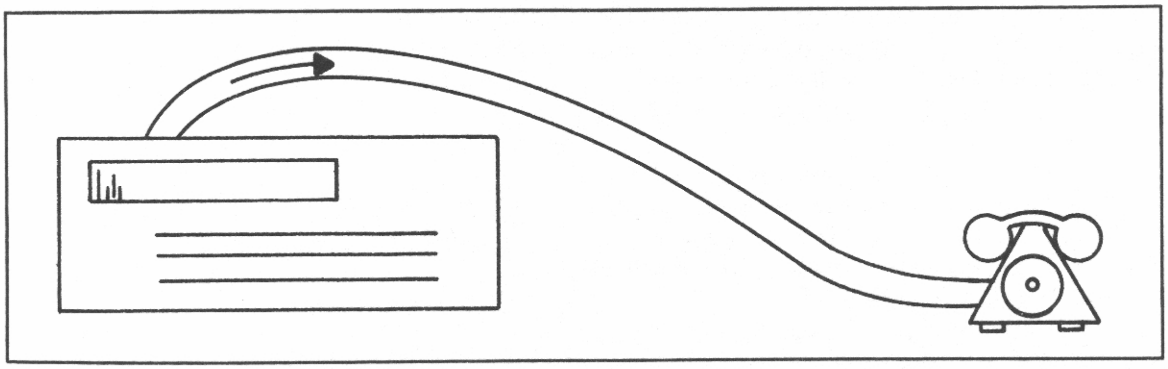


Fig. 12-2. Uploading is sending data from memory via the modem or RS-232.

puter B waits until Computer A says, "I'm finished." Then Computer B sends. Half duplex is slow. You use it when you have a bad connection and want to make sure no data is lost.

Originate/Answer

On the left side (as you face the display) of the Model 100 is an ANS/ORIG switch. This switch controls the modem. You will almost always leave the switch at ORIG.

It's really very simple. When you *initiate* the call through the modem to a computer, you set the switch at ORIG (originate). If you are expecting a call from another computer, you switch to ANS (answer). When set to ANS, the Model 100 will answer the call when it comes.

COMMUNICATIONS AND THE MODEL 100

As noted previously, with most computers you must buy additional hardware, such as a modem, and additional software, such as a communications package, to communicate with another computer. With the Model 100, all this is standard equipment. The Model 100 contains a built-in modem, RS-232 interface, and a communications program called TELCOM. You can also write BASIC programs for communications.

What doesn't come with the Model 100 are cables.

To use the modem, you need one of two ca-

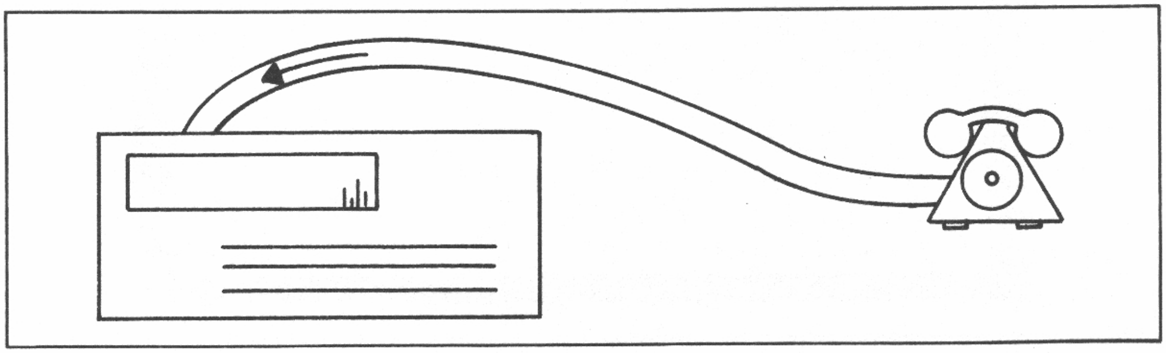


Fig. 12-3. Downloading is the term for receiving data via the modem or RS-232 and saving it in memory.

bles. If your phones have modular jacks, you can plug the modem in directly to your computer using the Model 100 Modem Cable. If your telephones do not have modular jacks, you must buy a TRS-80 Acoustic Coupler.

The Model 100 manual contains good instructions on plugging in the modem. Be sure to set the modem switch on the left side of the computer to DIR if you plug in directly. Set the switch to ACP if you use an acoustic coupler.

To use the RS-232 interface, you need Radio Shack's RS-232C Cable and an additional gizmo, called a null model adaptor. The adaptor is a plug that goes between the cable and one of the computers. All RS-232s are designed for modem communication. A null model adaptor "lies" to both computers and makes them believe that they are talking to each other through a modem.

To use TELCOM, you work with two modes, entry mode and terminal mode. Essentially, you use entry mode to establish communications. You use terminal mode to communicate.

ENTRY MODE

When you select TELCOM from the menu, you go directly into entry mode, which is sort of the "outer sanctum" of TELCOM. You use entry mode to dial the telephone and to prepare the Model 100 to go "on line."

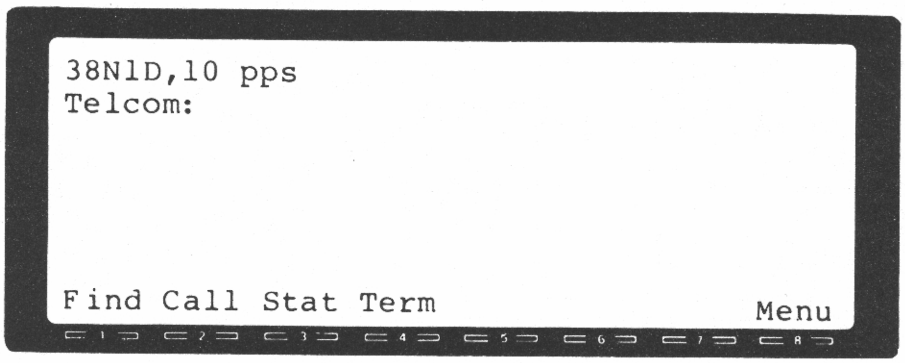
There are five operations you can do in entry mode. These are:

- ☐ Find a telephone number in ADDRSS's ADRS.DO file.
- ☐ Call a number.
- ☐ Set the communications parameters.
- ☐ Enter terminal mode.
- ☐ Return to the menu.

Each operation is performed with the function keys. The definition of the function keys in TELCOM's entry mode are given in Fig. 12-4.

F1 (Find) To Find a Telephone Number. To locate a specific telephone number in the ADRS.DO file, press the F1 (Find) function key and type a word to identify the telephone number you want.

F2 (Call) to Dial a Number. In the TELCOM



```
38N1D,10 pps
Telcom:




Find Call Stat Term                  Menu
```

Fig. 12-4. TELCOM's screen display of function key definitions in entry mode.

entry mode, the F2 (Call) key enables you to dial phone numbers. You can dial manually by typing the number from the keyboard, or you can dial automatically, letting the TELCOM program dial a number directly from the ADRS.DO file. You can also log on automatically to another computer. In order to use the Model 100 as an automatic dialer, you must first connect the Model 100's modem to the telephone.

F3 (Stat) to Set Communication Parameters. In order to communicate with another computer, you must make sure that both computers are using the same parameters as the other computer. In TELCOM entry mode, you press F3 (Stat) to set a wide range of parameters.

The manual is good on describing how to set the parameters, but one thing is not sufficiently emphasized. If you are using the modem, you must set the baud rate at M for modem.

To set the communications parameters

1. Select TELCOM from the menu.
2. Press F3 for STAT.
3. Type the parameters you want in the following order:

Baud (remember, type M for modem)
Word Length
Parity
Stop Bit
Line Status
Pulse Rate

For example, you type these parameters to prepare the Model 100 to communicate with CompuServe:

```
M7I1D,10
```

To communicate with the Model III via the RS-232 (using the Model III's default parameters), you type:

```
38N1E,10
```

Remember that the pulse rate `,10` is only used in the automatic dialing sequence.

4. Press the enter key.
5. Press the F3 key and the enter key.

F4 (Term) to Enter Terminal Mode. You press F4 after you have dialed—manually or automatically—a host computer's telephone number. The computer will automatically switch to the terminal mode.

F8 (Menu) to Return to the Menu. Pressing F8 exits the TELCOM program and returns you to the menu.

Manual Dialing

I don't know why you would want to dial manually. It's no faster than rotary dialing, and certainly slower than touch-tone dialing.

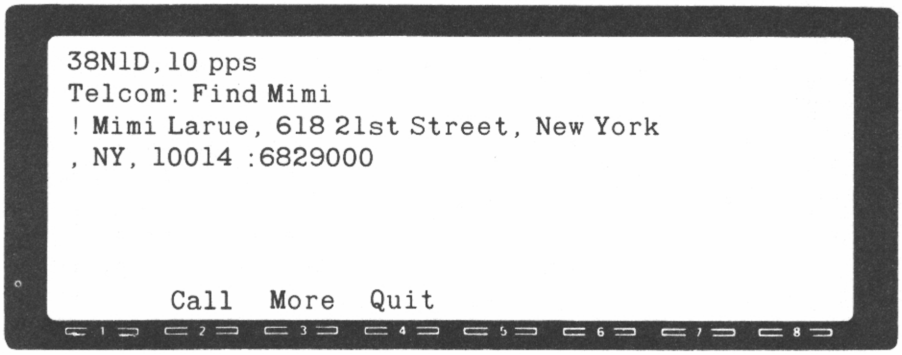
The procedure, should you want to use it, is as follows.

1. Select TELCOM from the menu.
The Model 100 enters entry mode.
2. Press the F2 key for Call.
3. Type the phone number and press the enter key.

The Model 100 dials the number. You must pick up the receiver before the computer finishes dialing.

Automatic Dialing

With the automatic dialing procedure, we're getting somewhere. TELCOM and ADRS.DO team up to form a very useful address book plus automa-



```
38N1D,10 pps
Telcom: Find Mimi
! Mimi Larue, 618 21st Street, New York
, NY, 10014 :6829000



     Call  More  Quit
```

Fig. 12-5. When dialing automatically, you enter a word to identify the telephone number you want to call. The computer searches the file and displays the telephone number.

tic dialer. You also use the automatic-dial feature to log on to information services. To use automatic dialing, you must first create the ADRS.DO file for the ADDRSS program, as described in Chapter 10.

To let the computer dial automatically, the procedure is as follows.

1. Select TELCOM from the menu.

The Model 100 enters entry mode.

2. Press the F1 key for Find.

The Model 100 displays **Find**.

3. Type an identification word for the number you want and press the enter key.

The Model 100 searches the ADRS.DO file for the identification word. When it finds it, the Model 100 displays the record (Fig. 12-5). If it finds more than one record, the Model 100 displays one record at a time. Press the F3 key to see the next record. Press F4 to quit.

4. When the Model 100 displays the number you want, press F2 for Call.

The Model 100 dials the telephone number. You must pick up the receiver before the computer finishes dialing.

TERMINAL MODE

You use entry mode to prepare for communications and to automatically dial and log on to another computer through the modem. Once you have telephoned another computer, you communicate with that computer in terminal mode. There are two ways to enter terminal mode from entry mode: automatically and manually. To enter

☐ Automatically, use F1 to find a number and F2 to log on.

☐ Manually, press F4 and dial the number.

There are six operations you can perform in terminal mode. These operations are controlled by six of the eight function keys (Fig. 12-6). In terminal mode, the definitions of the function keys will be defined at the bottom of the screen. To remove the definitions from the screen, simply press the label key.

The six operations performed in terminal mode are:

☐ Viewing previous lines on the display.

☐ Storing data in RAM files.

☐ Sending data in RAM to another computer.

☐ Switching between full duplex and half duplex.

☐ Printing data as it is received.

☐ Terminating communication with the other computer.

F1 (Prev). When you receive data on the display, the Model 100 displays eight lines at a time.

1. To see the previous eight lines, or full screenful, press the F1 key.
2. Press F1 again to continue.

F2 (Down). Use F2 to store (download) data in a RAM file as data is received.

1. Press F2.
The Model 100 prompts with File?
2. Type a name for a file to hold the data, and press the enter key.

The Model 100 stores the incoming data as a RAM document.

F3 (Up). Use F3 to send (upload) a document from RAM to another computer.

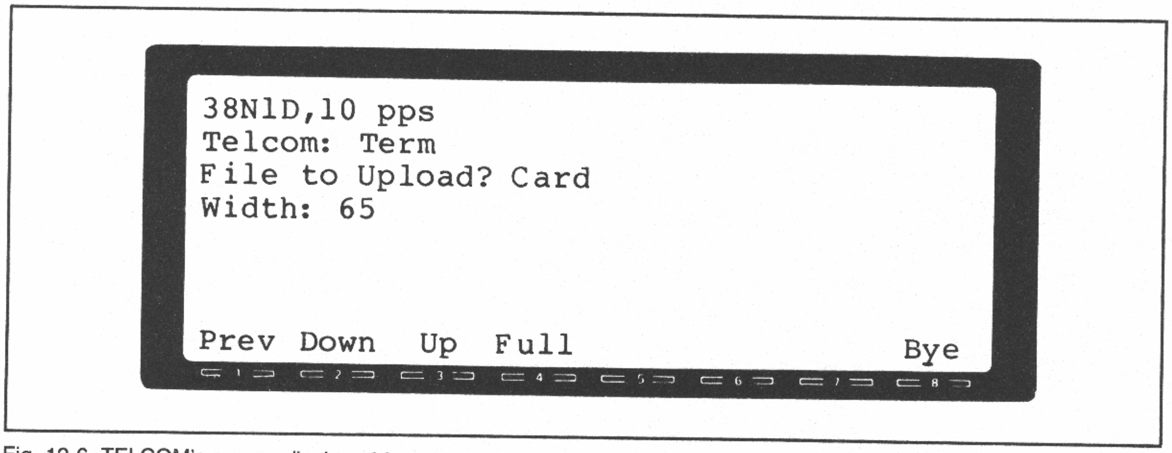
1. Press F3.
The Model 100 prompts with File?
2. Type the name of the RAM document file you want to send and press the enter key.
The Model 100 prompts with Width:
3. Type a number from 10 to 132 to indicate the number of characters per line to be printed and press the enter key.

The Model 100 sends a carriage return after the number of characters you type. For example, if you type 40, the Model 100 sends a carriage return after each group of 40 characters.

The Model 100 sends the file to the other computer.

F4 (Full). Press F4 to switch between full duplex and half duplex.

F5 (Echo). For some reason, the definition for this function key does not show up on the bottom of the



```
38N1D,10 pps
Telcom: Term
File to Upload? Card
Width: 65



Prev Down  Up  Full                 Bye
```

Fig. 12-6. TELCOM's screen display of function key definitions in terminal mode.

screen with the other function key definitions. When you press F5 in terminal mode, the computer "echos," that is transmits incoming data to both the screen and printer. Press F5 to send the data to the printer as it comes in. Obviously, you must have hooked up a printer.

F8 (Bye). Use F8 to terminate the communication. This is particularly important if you are logged on to a commercial data base that charges you by the minute, such as CompuServe.

1. Press F8.

The model 100 prompts with **Disconnect?**

2. Type Y and press the enter key or just press the enter key.

☐ Type Y and press the enter key to end the communication. The Model 100 disconnects the communications and returns to entry mode.

☐ Press only the enter key if you change your mind and decide not to disconnect. The Model 100 displays, **aborted.**

Chapter 13

Using the Modem

You use the Model 100's modem to communicate over the phone. There are two main uses for the modem.

- ☐ To communicate with commercial data bases such as CompuServe, Dow Jones, or The Source.
- ☐ To communicate with other users and your computer at home or the office.

In this chapter I will discuss one of the best-known information utilities, CompuServe. I'll show you how to use CompuServe. Once you understand how to communicate with one data base, you will quickly be able to expand your horizons and communicate with different services.

Before you use the modem, you must use the correct cables to plug the Model 100 into the phone as discussed in Chapter 12.

MODEM COMMUNICATIONS

To use the modem to communicate over the telephone, set the communication parameters, dial the other computer, then log on. Once you are logged on, you can "talk" to the other computer.

Many computers communicate over telephone lines. In fact, two major networks are used by almost all computers: Telenet and Tymnet, discussed at the end of this chapter. For now, let's take a look at the world that awaits you at the other end of your phone, including data bases and information utilities.

Data Bases

A data base is a collection of data. There are well over a thousand data bases that you can access via your modem. These data bases include encyclopedias, special-interest group data bases, and wire services. Some computer enthusiasts maintain their own independent data bases. All you need to access them is their telephone number.

Information Utilities

The commercial information utilities include collections of data bases and services. CompuServe, Dow Jones, and The Source are the major information utilities. You pay a per-minute fee for their use. They offer you a wide variety of information and services.

COMPUSERVE

CompuServe is a collection of data bases and services sold to users at the rate of 9 cents a minute. The service is available between 6:00 PM and 5:00 AM. In addition to its own services, CompuServe can connect you to individuals, other services, and other data bases such as the Associated Press Wire Service and the Washington Post. It's probably not an exaggeration to say that you can find out about almost *anything* from CompuServe or at least learn where to find out.

Okay, let's see how you use your Model 100 to join the brave new world of modem communications—specifically, CompuServe. To access this utility, you must follow these steps:

☐ Buy a CompuServe Start-Up Kit.

☐ Set up the Model 100 for modem communication with CompuServe.

☐ Log on.

The Start-Up Kit

Before you can use CompuServe, you must buy a CompuServe Start Up Kit. It costs approximately $19.95. The kit includes an identification number (called a user ID), a secret password, and the local CompuServe telephone number. Purchase of the kit also entitles you to one hour of use of CompuServe.

You can buy a CompuServe Start-Up Kit at any Radio Shack store. Do not let them sell you any additional software; TELCOM is all you need. If you cannot get to a Radio Shack store, call CompuServe toll free for the location nearest you: The telephone number is (800) 848-8990.

Setting Up the Model 100 for Communication with CompuServe

Once you have your CompuServe Start-Up Kit, you must set the Model 100's communication parameters so they are compatible with CompuServe. Follow these steps:

1. Select TELCOM from the menu.

The Model 100 enters entry mode.

2. Press F3 to select STAT to set the communication parameters.
3. Type the following CompuServe compatible parameters, and press the enter key.

```
M7I1D,10
```

These parameters represent:

M for modem (300 baud)
7 for 7-bit word
I for ignore parity
1 for 1 as the stop bit
D for disable wait (Xon/Off)
10 for ten pulses per second (dialer)

4. Press F3 and the enter key to lock in the settings.

The Model 100 displays the communication parameters:

```
M7I1D,10
```

LOGGING ONTO COMPUSERVE

Now you are all set; you are ready to log on.

You can log on to CompuServe manually, but the Model 100 provides you with a great automatic log-on capability. The timing of the commands in the manual sequence is tricky, and you may, therefore, not be successful on your first try. The automatic sequence is much easier to use and, therefore, recommended. If you understand the manual method, the automatic method will make more sense, so let me show you how to log on manually before I show you how to do it automatically.

Logging On Manually

The basic idea is to use TELCOM's entry mode to dial the telephone number. Then you switch to terminal mode, send a CTRL C to say "hi!" then type your identification number and password. Here is the step-by-step procedure.

1. Select TELCOM from the menu.

TELCOM goes into entry mode.

2. Press F2 for Call.

TELCOM prompts with **Call**.

3. Type CompuServe's local telephone number and press the enter key.

TELCOM dials the number.

4. Wait a minute for communication to be established, then press F4 to enter terminal mode.

The Model 100 enters terminal mode.

5. Hold down the control (CTRL) key and type **C** to begin the log-on procedure.

CompuServe asks for **User ID**:

6. Type your user ID and press the enter key.

CompuServe asks for your **Password**:

7. Type your password and press the enter key. (For security reasons, your password will *not* be displayed as you type it.)

CompuServe displays **CompuServe Information Service**.

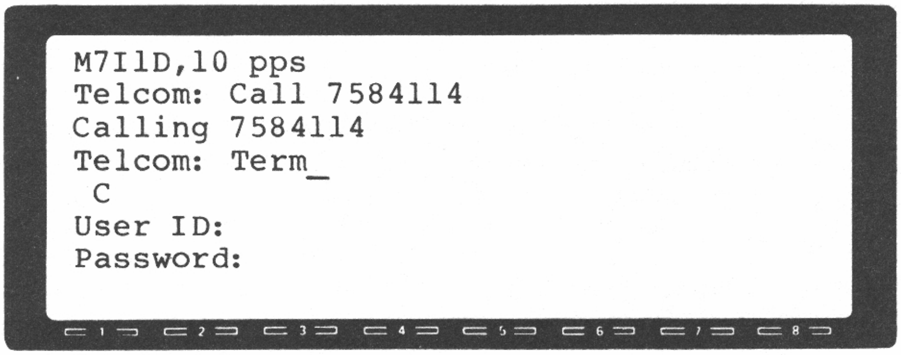
You are now logged on (Fig. 13-1). Until you subscribe, CompuServe asks you to sign on each time you log on.

Logging On Automatically

As just illustrated, you follow this sequence of steps to log on to CompuServe:

1. Use TELCOM to dial the number. Enter terminal mode.
2. Type a **CTRL C**.
3. Type your user ID.
4. Type your password.

To log on automatically, you first store this



```
M7I1D,10 pps
Telcom: Call 7584114
Calling 7584114
Telcom: Term_
 C
User ID:
Password:
```

Fig. 13-1. Manual log-on display.

:5551223<=^C?U11111,111^M?PCIRCUS*UNION^M>:

Item	*Description*
:	You must enclose the entire thing—number and sequence—within colons.
5551234	Your local CompuServe number.
<	You tell the Model 100 you are giving a log-on sequence by enclosing it in <>.
^C	Send CTRL C to CompuServe, in other words, saying, "I am ready to log on."
?U	Wait for CompuServe to send a U—user ID.
11111,111	Your user ID.
^M	ENTER.
?P	Wait for CompuServe to send a P—password.
CIRCUS*UNION	Your password.
^M	ENTER.
>	You tell the Model 100 this is a log-on sequence by enclosing it in <>.
:	You must enclose the entire thing—number and sequence—within colons.

Fig. 13-2. The log-on sequence for communicating with CompuServe.

sequence in the ADRS.DO file. Then use TELCOM's auto-dial feature to call CompuServe. TELCOM executes the sequence automatically. Well, it's not quite that simple; you must include some codes in the sequence. The codes are

< > You must enclose the sequence within these symbols.

^ This code (shift 6) sends the next character as a "control" character. For example C sends CTRL C.

? Wait for a character before continuing. For example, type ?U to wait for a U.

Cancel that code! If you want to send ? and not use ? as a code, type !?.

The manual does a good job of explaining the log-on sequence, but here it is again.

1. Open the ADRS.DO document.
2. Press the control key and the down arrow cursor key to move to the end of the document.
3. Type the CompuServe record as follows:

☐ Type a name to identify the log-on sequence. For example, type CIS for CompuServe Information Service.

☐ Type the log-on sequence (Fig. 13-2) for CompuServe as follows.

```
:5551223⟨=∧C?U11111,111∧M?
   PCIRCUS∗UNION∧M⟩:
```

☐ Press the enter key to end the record.

4. The whole thing looks like this:

```
    CIS :5551223⟨=∧C?
U11111,111∧M?PCIRCUS*
        UNION∧M⟩:
```

5. Press F8 to return to the menu.

Using the Log-On Sequence

Once you have typed the automatic log-on sequence in your ADRS.DO file, it's remarkably easy to log on. The steps are

1. Select TELCOM from the menu.
2. Press F1 for Find.
3. Type the CompuServe ID (CIS), and press the enter key.

The Model 100 finds the CIS record and displays it.

4. Press F2 for Call.

The Model 100 executes the log-on sequence (Fig. 13-3) as follows:

```
:5551223⟨=∧C?U11111,111∧
 M?PCIRCUS*UNION∧M⟩:
```

The display is given in Fig. 13-4. Note: for security reasons, your password will *not* be displayed as it is being transmitted to CompuServe.

USING COMPUSERVE

Each data base or information utility has its own protocol for locating data or a particular service. CompuServe provides two ways to locate the data or service you want. They are menus and code numbers.

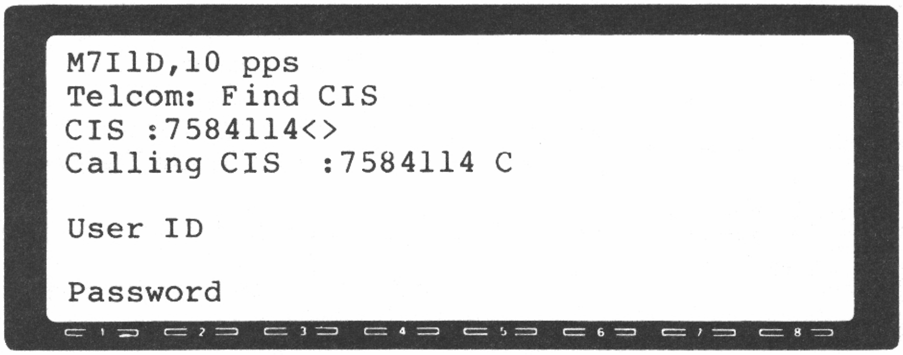
When you first log on, CompuServe displays the main menu. From the main menu you can select a menu option and work your way through a series of menus to the service or data you want.

If you know the page number of the data or service you want, you type GO, the code number

:5551223<=∧C?U11111,111∧M?PCIRCUS*UNION∧M>:

Item	*Description*
5551223	It calls the number.
<	It's an automatic log-on sequence.
=	Wait 2 seconds.
∧C	Send a CTRL C.
?U	Wait for U(user ID).
11111,111∧M	Send your ID and ENTER.
?P	Wait for P (password)
CIRCUS*UNION	Send your password.
∧M	Send ENTER.
>	That's it; you are logged on.

Fig. 13-3. Execution of the CompuServe log-on sequence.



```
M7I1D,10 pps
Telcom: Find CIS
CIS :7584114<>
Calling CIS  :7584114 C

User ID

Password
```

Fig. 13-4. Automatic log-on display.

you want, and press the enter key. For example, to play backgammon, you type GO GAM-31.

CompuServe sends you its index of code numbers when you sign on. You can display the index by typing GO IND.

Using TELCOM's Features

Once you log on, TELCOM's terminal mode enables you to take full advantage of the data base or information utility. The operations you can perform are:

- ☐ View previous lines
- ☐ Download
- ☐ Upload
- ☐ Echo
- ☐ Disconnect

You can, of course, use these features with almost any data base information service, but I'll continue to use CompuServe as an example. The operations are explained as follows.

F1 (Prev). To see the previous eight lines, press F1. To continue, press F1 again.

F2 (Download). Download helps you take full advantage of data bases. You log on, go straight to the data you want, download it, and log off. Then you can examine the data at your leisure.

For example, let's say you want the latest baseball scores from CompuServe. The code number for baseball scores is APN-16. The procedure is

1. Log on.
2. Press F2 for download, type the name of the file (BASBAL), and press the enter key.
3. Type the item code number (GO APN-16) and press the enter key.

CompuServe sends the scores from the Associated Press Wire Service. The Model 100 downloads (stores) the data in BASBAL.DO.

4. After the data is received, press F8 to disconnect.

Now the scores are stored in the BASBAL.DO file. Use TEXT at your leisure to display them. You can also write a BASIC program that logs on, gets the scores, downloads them, and logs off.

F4 (Upload). To send a file from RAM, use upload.

The Model 100 sends the data through the model to the other computer. For example, to post a message on CompuServe's bulletin board, use TEXT to compose the message and edit it. Call the message file MESG.DO. Then log on, go to the bulletin board, and upload MESG.DO. Step by step the procedure is:

1. Use TEXT to type your message. Name the document MESG.DO.
2. Use TELCOM to log on to CompuServe. Then type GO Bulletin Board.
3. When CompuServe asks for the message, press F3 for upload.
   The Model 100 asks File to upload?
4. Type MESG.DO and press the enter key.
   The Model 100 prompts with Width:
5. Type a number for the number of characters per line you want displayed. The number can be from 10 to 132. Press the enter key.

The Model 100 sends the MESG.DO document.

F5 (Echo). This function key enables you to send data to the printer as the data is received. Personally, I prefer to download and use TEXT to edit the data before I print it. However, to send data to the printer as it comes in, press F5.

Getting Your Money's Worth

Since you pay 9 cents per minute, you should

☐ Log on

☐ Go to the data.

☐ Download the data.

☐ Disconnect.

The key here is *download.* When you use TELCOM's download feature, the Model 100 stores the data in RAM as it comes in. Then, at your leisure, you can display, print, or upload the data.

Finding the Data or Service You Want

If you have never used CompuServe before, the vast number of possibilities it offers you can be overwhelming. Ultimately, you are just going to have to experiment.

You can work you way through a series of menus, but it's best to go straight to the data or service you want. These codes are listed in the CompuServe Index. Each data base and service has a code. For example, baseball scores are coded APN-16; the astology game is GAM-45. The CompuServe index is an alphabetical listing of all the data and services offered with the code for each. One of the best ways to get the most out of CompuServe is to download the index when you first log on.

Once you know the code for the data base or service you want, you can go directly to it by typing GO, the code, and pressing the enter key. For example, to see your horoscope after you log on, type:

GO GAM-45

To download the Index, the steps are

1. Log on to CompuServe.
2. Press F2 for download.
   The Model 100 displays Name of file.
3. Type INDEX. Press the enter key.
4. Type GO IND and press the enter key.
   CompuServe sends the index, and the Model 100 downloads it into RAM.
5. When the transmission is complete, press F8 and type Y to disconnect.
6. To view the index, just select INDEX.DO from the menu.

TELENET AND TYMNET

Telenet and Tymnet are both networks that make it possible for you to call a local number to talk to a data base. The large data services—CompuServe, Dow Jones, and The Source—use these systems. Most services—public and private—use Telenet or Tymnet for communications.

In addition, each network will provide you with telephone numbers of data bases, users, and services in your area.

To call Telenet dial (800) 336-0437. In Virginia the telephone number is (800) 572-0408.

To call Tymnet in the East dial (800) 336-0149. In the West dial (800) 323-7389.

Chapter 14

Using the RS-232

You use the Model 100's RS-232 interface to communicate directly with another device. The device can be another computer, a disk drive, or a printer. It can also be an appliance module such as would be used to automatically turn on your coffee maker in the morning or turn lights off. Nine times out of ten you will use the RS-232 to transmit data back and forth from the Model 100 to another computer.

For example, I used the RS-232 to transmit many of the figures in this book from my Model 100 to my Model III, and printed them on my Daisy Wheel II printer.

To use the RS-232 to communicate between the Model 100 and another computer, you must be sure that

☐ Both computers are equipped with an RS-232 interface.

☐ The Model 100 is connected to the other computer with an RS-232 cable and a Null Modem Adaptor.

☐ Both computers use the same communication parameters: baud, word length, parity, stop bits, and wait mode.

☐ Both computers are programmed to send and receive data over the RS-232. The Model 100's TELCOM program enables you to send and receive over the RS-232. The other computer's program software must also be programmed.

The Model 100 user's manual provides programs for transferring data between the Model 100 and the TRS-80 Model II computer.

In this chapter, I will explain how to use the RS-232 to communicate between the Model 100 and the TRS-80 Model III Computer. Even if you don't have a Model III, the description will provide you with a general overview of how to transfer data between two computers with the RS-232. If you own a computer with an RS-232 interface (other than the Model II or Model III) consult your owner's manual for information on programming the RS-232.

CONNECTING THE COMPUTERS

Obviously, before you communicate over the RS-232, you must use an RS-232 cable to connect the two computers. Moreover, you must also use a null modem adaptor. In addition, you must set both computers to the same communications parameters.

When you program the Model III to send and receive data over the RS-232, you can set the parameters as part of the program. On disk systems you can use the TRSDOS SETCOM command to set the parameters. For our purposes, I will show you how to set the Model 100 to match the Model III's default parameters. Once you get the hang of it, you can try out different settings.

The default RS-232 settings for the Model III are:

Baud: 300
Word Length: 8
Parity: None
Stop bits: 1
Wait: Enable

So you must set the Model 100's parameters as 38N1E (Fig. 14-1).

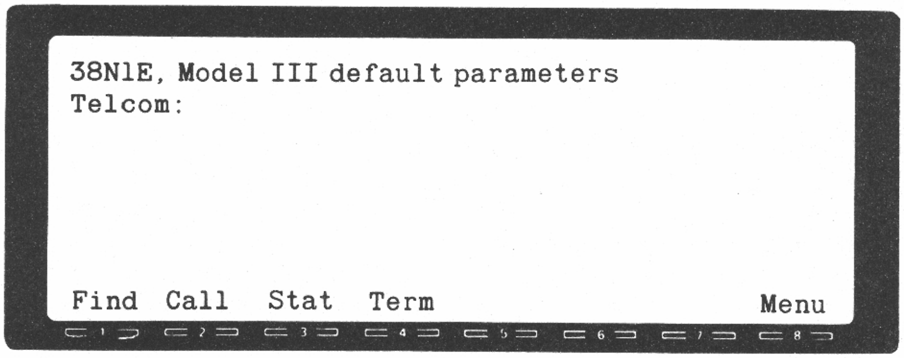
To set the Model 100 to these parameters, the steps are:

1. Select TELCOM from the menu.
2. Press F3 for STAT (communications parameters).
3. Type 38N1E and press the enter key.
4. Press F3 and the enter key.

The Model 100 and the Model III will now be set with the same communications parameters.

PROGRAMMING THE MODEL III FOR COMMUNICATIONS

The Model 100 comes with the TELCOM communications program built in. However, to use the Model III's RS-232, you must write a BASIC



```
38N1E, Model III default parameters
Telcom:
```

Fig. 14-1. The Model 100 with Stats communications parameters to match the Model III's default parameters.

program. You can buy a commercial program (VIDTEX), but why bother? It's pretty easy to write a BASIC program.

Model III BASIC and the RS-232

The Model III's ROM (read only memory) has three subroutines that "work" the RS-232. They are

☐ **$RSINIT** at byte address 90: This routine initializes the RS-232.

☐ **$RSTX** at byte address 85: This routine *sends* a character out the RS-232.

☐ **$RSRCV** at byte address 80: This subroutine *receives* a character from the RS-232.

To use these subroutines in a BASIC program, you must use the Model III's DEFUSR (define user) and USR (user) commands.

To define a user function the form is:

DEFUSR0 = the address of the machine language routine

For example

```
DEFUSR0 = 90
```

To call the user function you defined as 0, type

```
X=USR0(0)
```

In the initialize section of the program, define three user functions as 0, 1, and 2 as follows

```
DEFUSR0 = 90  'Initialize RS-232
DEFUSR1 = 85  'Send a character over the
               RS-232
DEFUSR2 = 80  'Invoke USR2 to receive a
               character
```

Also in the initialization section of the program, call the user function you defined to initialize the RS-232:

```
X=USR0(0)
```

Sending a Character

To send a character over the RS-232, first poke the ASCII code for that character into the RS-232 "waiting room" byte address 16880. Then call the ROM subroutine you defined as 1. For example, these lines send the letter A over the RS-232:

```
CD = ASC("A")   'Get the ASCII code for A
POKE 16880,CD   'Poke the ASCII code into the
                 RS-232 output buffer—byte
                 16880
X=USR1(0)       'Call the subroutine defined as
                 1—the subroutine sends a
                 character out the RS-232.
```

Receiving a Character

To receive a character over the RS-232, call the ROM subroutine you defined as 2. Then peek into the RS-232 "receiving room," byte address 16872.

For example, these lines receive a character from the RS-232:

```
X=USR2(0)          'Call the subroutine defined
                    as 2—the subroutine
                    receives a character from
                    the RS-232 and places it in
                    byte 16872.
CD=PEEK(16872)     'See what the subroutine
                    left in byte 16872.
RC$=CHR$(CD)       'Convert the number found
                    in the buffer to a character.
PRINT RC$;         'Display the character.
```

That's the BASIC idea of how you program the Model III to transmit over the RS-232.

UPLOADING FROM THE MODEL 100

Since the Model 100 is already programmed for communications with the TELCOM program, you have much less work to do, than with the Model III.

Let's say you want to upload or send the SALES.DO letter from the Model 100 to the Model III, and you have already connected the two computers, set the parameters, and programmed the Model III for communications. The procedure for uploading is

1. Run the Model III download program.
2. Select TELCOM from the Model I menu.
3. Press F4 to enter terminal mode.
4. Once in terminal mode, press F3 to select upload.

The Model 100 prompts with

File to upload?

5. Type the name of the RAM file and press the enter key.

The Model 100 prompts with

Width:

6. Press the enter key or type a number from 10 to 132 to set the number of characters per line and press the enter key.

The Model 100 sends the file from RAM, over the RS-232, to the Model III.

MODEL III PROGRAMS FOR COMMUNICATING WITH THE MODEL 100

Here is a brief summary of the six Model III RS-232 programs.

Comfile. This program (Fig. 14-2) downloads a file from the Model 100. The Model III receives the file, stores it in high memory, and writes the file to disk.

UPM1. This program (Fig. 14-3) uploads a file to the Model 100. The Model III loads an ASCII file from disk and sends it to the Model 100.

Term. This program (Fig. 14-4) converts the Model III to a terminal.

Comread. This program (Fig. 14-5) reads an ASCII file from disk and displays it on the screen.

Comprint. This program (Fig. 14-6) reads an ASCII file from disk and prints it.

Menu. This program (Fig. 14-7) acts as a communications menu so you can easily select the Model III communications program you want.

Fig. 14-2. Model III Comfile program.

```
100 DEFINT A-Z:CLS
110 PRINT TAB(15) "******** DISPLAY AND SAVE INPUT ********"
120               'Set up RS-232 USR routines, and initialize
130 POKE 16890,1
140 POKE 16888,(5*16)+5
150 DEFUSR0= 90
160 X=USR0(0)
```

```
170 DEFUSR1 = 80
180 DEFUSR2 = 85
190 CI = 16872
200 CO = 16880
210 POKE CO,19:X=USR2(0)
220 ST=-15000
230 PRINT
240 LINE INPUT "To what file shall I write the input? ";F$
250 '
260 ' Receive from the Model 100 and poke ASCII codes into
270 ' high RAM
280 '
300 PRINT:PRINT "PRESS ANY KEY WHEN READY": PRINT "(Press the Mod
el 100's ESC key when the upload is complete.)"
310 IR$=INKEY$: IF IR$="" THEN 310
320 CLS
330 POKE CO,17:X=USR2(0):POKE CO,19
340 FOR ST=-15000 TO 0
350 X=USR1(0)
360 A=PEEK(CI):POKEST,A:PRINTCHR$(A);
370 IF A = 27 THEN I = ST:GOTO 500
380 NEXT
390 X=USR2(0)
480 '
490 'Peek at ASCII codes, and write characters to disk
500 '
510 PRINT:PRINT "I'M WRITING THE DATA TO DISK. PLEASE WAIT A MOME
NT."
520 OPEN "O",1,F$
530 I2=I: FOR I = -15000 TO I2
540 A=PEEK(I):PRINT#1,CHR$(A);
550 IF A = 27 THEN 570
560 NEXT I
570 CLOSE:END
580 CLOSE
590 PRINT:PRINT "End of communication.": PRINT "Do you want to sa
ve anything else (Y/N)"
600 IR$ = INKEY$: IF IR$ = "" THEN 600
610 IF IR$ = "N" THEN RUN"MAM"
620 IF IR$ = "Y" THEN GOTO 100
630 GOTO 320
```

Fig. 14-3. Model III UPM1 program.

```
100 CLEAR 10000:DEFINT A-Z
110 CLS:PRINTTAB(15) "******** LOAD A FILE TO THE MODEL 100 *****
***"
```

```
120 POKE 16890,1'SET PARAMETERS
130 POKE 16888,(5*16)+5  'SET PARAMETERS
140 DEFUSR0=90' Define USR 0 to initialize RS-232
150 X=USR0(0)' Initialize RS-232
160 DEFUSR1 = 80' Define USR 1 to receive a character
170 DEFUSR2 = 85' Define USR 2 to send a character
180 CI = 16872' Set "CI" as address of input buffer
190 CO = 16880' Set "CO" as address of output buffer
200 PRINT:PRINT:LINE INPUT "What file do you want to upload? ";F$
210 OPEN "I",1,F$
220 PRINT:PRINT:PRINT "Okay.  Press any key when the Model 100 is
 ready."
230 IR$=INKEY$: IF IR$="" THEN 230
240 IF EOF(1) THEN 340
250 LINE INPUT#1, A$
260 LI = LEN(A$)' These lines send one character at a time
265 IF LI = 0 THEN 240
270 FOR I = 1 TO LI
280 A = ASC(MID$(A$,I,1))
290 POKE CO,A: X = USR2(0)' Send the character out the RS-232
300 NEXT I
310 POKE CO,13: X=USR2(0)' Send a carriage return
320 POKE CO,10: X=USR2(0)
330 GOTO 240
340 CLOSE
350 CLS:PRINT "I'm finished.  Do you want to upload another file
(Y/N)?"
360 IR$=INKEY$:IF IR$="" THEN 360
370 IF IR$="Y" THEN 100
380 RUN"MAM"
```

Fig. 14-4. Model III Term program.

```
100 DEFINT A-Z
110 CLS:PRINT "THE MODEL III IS NOW A TERMINAL"
120 POKE 16890,0
130 POKE 16888,(5*16)+5
140 DEFUSR0= 90
150 X=USR0(0)
160 DEFUSR1 = 80
170 DEFUSR2 = 85
180 CI = 16872
190 CO = 16880
200     'Look for a character from the Model 100
210 X=USR1(0)
220 C$ = CHR$(PEEK(CI))
```

```
230 PRINT C$;
240     'Look for a character from the keyboard
250 C$ = INKEY$
260 IF C$ = "" THEN 200
270 PRINT C$;
280 POKE CO, ASC(C$)
290 X=USR2(0)
300 GOTO 200
```

Fig. 14-5. Model III Comread program.

```
100 DEFINTA-Z
110 CLEAR 2000
120 CLS:PRINTTAB(15) "******** READ AN ASCII FILE ********":PRINT
130 LINE INPUT "What file do you want to display? ";F$: PRINT
140     'Open the ASCII file you want to display
150 OPEN "I",1,F$
160 FOR I = 1 TO 10000
170 IF EOF(1) THEN 210
180     'Input each line from the file and display it.
190 LINEINPUT#1,C$:PRINTC$
200 NEXT
210 CLOSE
220 PRINT:PRINT "Do you want to display another file (Y/N)?"
230 IR$=INKEY$:IF IR$="" THEN 230
240 IF IR$ ="Y" THEN 100
250 RUN"MAM"
```

Fig. 14-6. Model III Comprint program.

```
100 CLEAR 5000: PS=16425:POKE PS,1' Set the line counter to 1
110 CLS:PRINTTAB(15) "******** PRINT A FILE ********"
120 PRINT:LINE INPUT "What file do you want to print? ";F$
130     'Open the ASCII file you want to print
140 OPEN "I",1,F$
150     'Input each line and print it
160     '(the TAB(10) creates a left margin)
180 IF EOF(1) THEN 300
190 LINEINPUT#1,C$
```

```
 200      'If the line is longer than 65 characters, divide it
 210 LS = LEN(C$): IF LS<66 THEN LPRINT TAB(10) C$:GOTO 180
 220 RS = LS-65: LF$=LEFT$(C$,65):RT$=RIGHT$(C$,RS)
 230 LPRINT TAB(10)LF$:LPRINT TAB(10)RT$
 240      'On the 54th line, print another page
 250 X=PEEK(PS): IF X = 54 GOSUB 1000
 260 GOTO 180
 270 END
 290     'END OF JOB
 300 CLOSE
 310 X = PEEK(PS): Y=66-X
 320 FOR I = 1 TO Y:LPRINT:NEXT' Eject last page
 330 CLS:PRINT:PRINT "DO YOU WANT TO PRINT ANOTHER FILE (Y/N)?"
 340 IR$=INKEY$:IF IR$="" THEN 340
 350 IF IR$="Y"GOTO 100
 360 RUN"MAM"
1000 '
1010            'NEW PAGE SUBROUTINE
1020 '
1030 CLS:PRINT:PRINT "Do you want to print another page (Y/N)? "
1040 IR$=INKEY$:IF IR$="" THEN 1040
1050 IF IR$="N" THEN RUN"MAM"
1060 FOR I2=1 TO 20:LPRINT:NEXT' EJECT PAPER
1070 POKE PS,1
1080 PRINT:PRINT:PRINT "PRESS ANY KEY TO CONTINUE."
1090 IR$=INKEY$:IF IR$="" THEN 1090
1100 RETURN
1110 END
```

Fig. 14-7. Model III Menu program.

```
100 'MASTER COM MENU
110 CLS
120 PRINT:PRINT TAB(15) "******** COMMUNCATIONS ********"
130 PRINT:PRINT
140 PRINT TAB(15) "1.....Save M100 Input"
150 PRINT TAB(15) "2.....Display File"
160 PRINT TAB(15) "3.....Output TO the M100"
170 PRINT TAB(15) "4.....Use the MIII as a Terminal"
180 PRINT TAB(15) "5.....Print a File"
190 PRINTTAB(15) "6.....TRSDOS"
200 PRINTTAB(15) "7.....BASIC"
```

```
 210 PRINT TAB(15) "8.....DIRECTORY"
 240 PRINT:PRINTTAB(15) "Choose your selection... ";
 250 PRINT TAB(40) CHR$(14);
 260 IR$=INKEY$:IF IR$="" THEN 260
 270 PRINTIR$;
 280 X=VAL(IR$)
 290 ON X GOTO 1000,2000,3000,4000,5000,6000,7000,8000
 300 GOTO 10
1000 RUN"COMFILE"
2000 RUN"COMREAD"
3000 RUN"UPM1"
4000 RUN"TERM"
5000 RUN"COMPRINT"
6000 CMD"S"
7000 NEW
8000 PRINT:PRINT:PRINT "Which directory 0 or 1?"
8010 IR$=INKEY$: IF IR$="" THEN 8010
8020 DN = VAL(IR$): IF DN > 1 THEN 8000
8030 IF DN = 0 THEN CMD"D:0" ELSE CMD"D:1"
8040 PRINT:PRINT "Press any key to return to Menu."
8050 IR$ = INKEY$: IF IR$="" THEN 8050
8060 GOTO 100
```

PART 5
BASIC

The BASIC interpreter designed by Microsoft for the Model 100 is, let's not mince words, spectacular. In fact, the heart and soul of the Model 100 is its BASIC program. You can use it to write an endless variety of programs in Microsoft BASIC for game playing, account management, budgeting, keeping track of inventory, and much more.

In addition, you can use the built-in BASIC program to control the Model 100. That is, you can use BASIC to kill, append, and name (referred to as file management) your RAM files; to set the time and date; to control start-up of the machine, and to define the function keys. These are often referred to as "housekeeping" functions.

Finally, you can use the built-in BASIC program to enhance TEXT, SCHEDL, ADDRSS, and TELCOM. By "enhance" I mean you can use BASIC to write programs that will help you better or more efficiently communicate with other computers with TELCOM; run, load, save, and edit documents with TEXT; and sort data bases with ADDRSS and SCHEDL.

Part 5 does not teach the BASIC language. I assume that you have at least a nodding acquaintance with BASIC. If you do not know the basics of the BASIC language, I suggest you read *Getting Started with TRS-80 BASIC,* (available from Radio Shack, catalog number 26-2107) before you read Part 5. The manual that comes with the Model 100 provides a comprehensive list of all commands and functions, so they are not all discussed here. This part, therefore, only covers special—or particularly helpful—Model 100 BASIC commands and functions.

Chapter 15

Introduction

Before we delve into the particulars, this chapter gets the basics of BASIC out of the way. These include how to invoke BASIC from the menu, how to run a previously written BASIC program from the menu, and how to use the function keys in BASIC.

INVOKING BASIC

To use Model 100 BASIC, you must first "invoke" or enter it. You invoke BASIC from the menu the same way you invoke the other programs.

To Invoke BASIC:

☐ Display the menu.

☐ Position the cursor on BASIC (Fig. 15-1) and press the enter key.

The Model 100 displays:

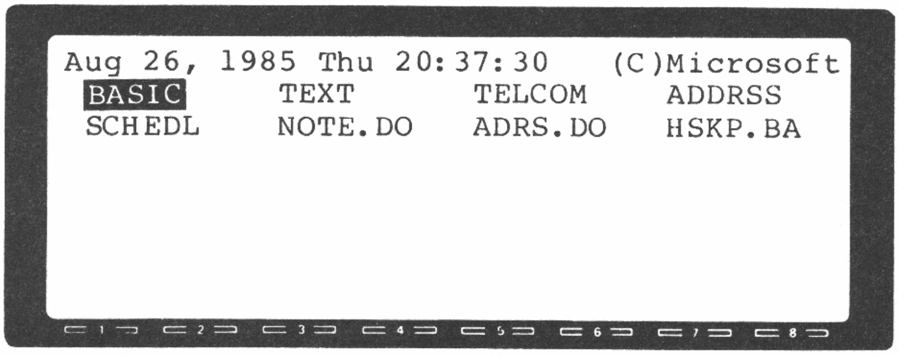
```
TRS-80 Model 100 Software
Copr. 1983 Microsoft
##### Bytes free
Ok
```

BASIC is invoked. You can type a program or type BASIC commands directly.

RUNNING A BASIC PROGRAM FROM THE MENU

When you save a BASIC program in RAM (from BASIC, the cassette, the modem, or from the RS-232), the Model 100 lists it in the menu. For example, if you write a program to find square roots and save it in RAM as ROOT, the Model 100 appends a .BA suffix to the program name and lists it in the menu as ROOT.BA. You can run the program directly from the menu without first invoking BASIC. The procedure is

1. Display the menu.



```
Aug 26, 1985 Thu 20:37:30   (C)Microsoft
 BASIC     TEXT      TELCOM    ADDRSS
 SCHEDL    NOTE.DO   ADRS.DO   HSKP.BA
```

Fig. 15-1. To select the BASIC program from the menu, position the cursor over BASIC and press the enter key.

2. Position the cursor on the BASIC program (Fig. 15-2) you want to run and press the enter key.

The Model 100 runs the BASIC program.

When you invoke BASIC, you can type *direct* commands or programs. To type a BASIC program, you must type a line number before each command. For example, type the following short program.

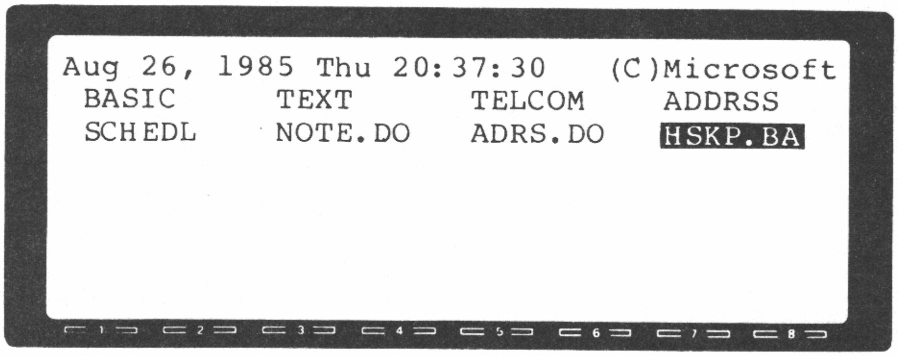
```
160  CLOSE 1
170  KILL "SORT.DO"
180  FILES
```

BASIC executes the commands when you ***run*** the program. If you run this short program; BASIC will delete the SORT.DO document when it reaches line 170.

If you type a command directly, that is with no number, and press the enter key, BASIC executes the command immediately. For example, if you type

```
KILL "SORT.DO"
```

and press the enter key, BASIC will kill (delete) the SORT.DO document immediately.



```
Aug 26, 1985 Thu 20:37:30   (C)Microsoft
 BASIC     TEXT      TELCOM    ADDRSS
 SCHEDL    NOTE.DO   ADRS.DO   HSKP.BA
```

Fig. 15-2. To run a BASIC program from the menu, position the cursor over the program name and press the enter key.

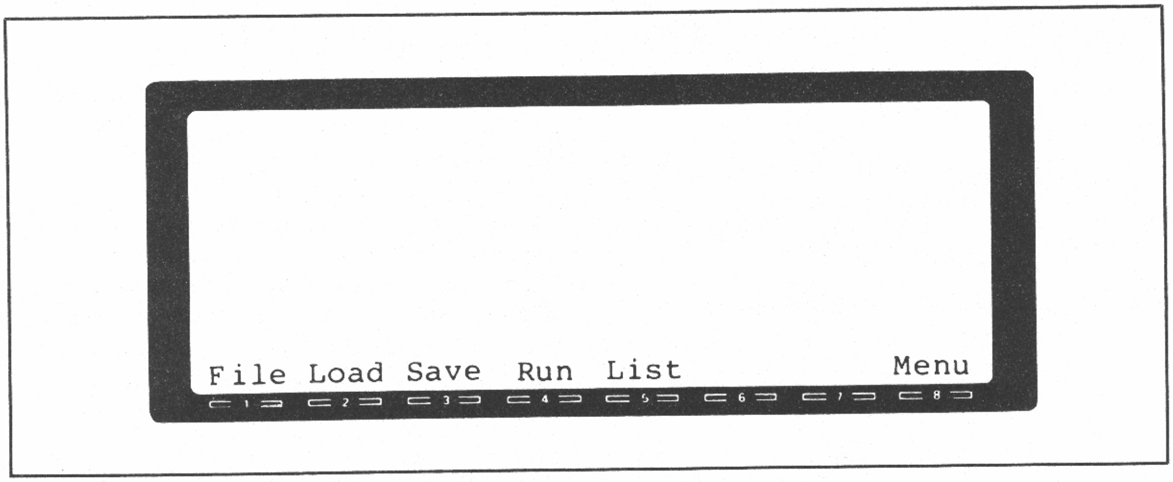


Fig. 15-3. Press the label key to display the function key functions in BASIC.

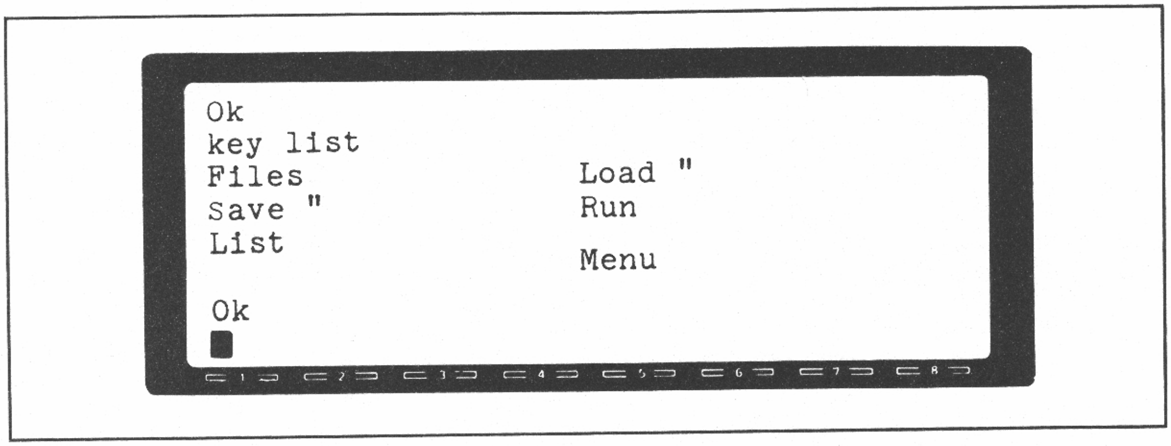
USING THE FUNCTION KEYS IN BASIC

In BASIC, function keys F1 through F5 and F8 are defined to perform the more common commands you use to work with BASIC programs. You use them to do things like load, run, and list programs. You can also define the function keys yourself as explained in Chapter 18.

To display the function key definitions, press the label key, the definitions will appear at the bottom of the screen (Fig. 15-3). Another way to display the definitions is to type key list. This is a BASIC command. The definitions will appear listed on the display (Fig. 15-4).

F1 (File). Press F1 to display the directory of RAM files. You can also display the directory by typing FILES and pressing the enter key.

F2 (Load). Press F2 to load a BASIC program. An



```
Ok
key list
Files                  Load "
Save "                 Run
List
                       Menu

Ok
█
```

Fig. 15-4. Use the key list command to display the function key functions in a list.

alternative method is to type LOAD ". After the Model 100 displays

```
LOAD"
```

type the name of the program, type a closing quotation mark ("), and press the enter key, as in

```
LOAD "ROOT"
```

F3 (Save). Press F3 to save a BASIC program. An alternative method is to type SAVE ". When you see the display

```
SAVE"
```

type the name of the BASIC program you want to save, type a closing quotation mark ("), and press the enter key,

```
SAVE "ROOT"
```

F4 (Run). Press F4 to run the resident BASIC program. An alternative method is to type RUN and press the enter key.

F5 (List). Press F5 to list the resident BASIC program. An alternative method is to simply type LIST and press the enter key. The Model 100 will list the program on the display. To hault the scrolling of the program lines off the screen, press the pause key. Press the pause key a second time to resume the listing of program lines.

F8 (Menu). Press F8 to exit BASIC and return to the menu. You can also return to the menu by typing MENU and pressing the enter key.

Chapter 16

Housekeeping

The Model 100's BASIC enables you to perform certain housekeeping chores such as cleaning up the menu (deleting and renaming RAM files), controlling the power, controlling and checking memory, setting the time and date, and so on. This chapter discusses the BASIC commands you will find useful for these Model 100 housekeeping chores:

Date and Time Commands
Power and Start-Up Commands
Commands to Work with the Function Keys
Commands to Manage RAM Files
Commands to Manage Memory

You will find specific programs in Chapter 19.

DATE$ AND TIME$

The Model 100 contains a "real time" clock that runs continuously. The menu displays the date, the day of the week, and the time (Fig. 16-1).

The BASIC constants time$ and date$ enable you to *retrieve* the time and date or to *set* them.

Date$ stores the date as a string variable in the format MM/DD/YY. For example, July 4, 1985 is stored in date$ as 07/04/85.

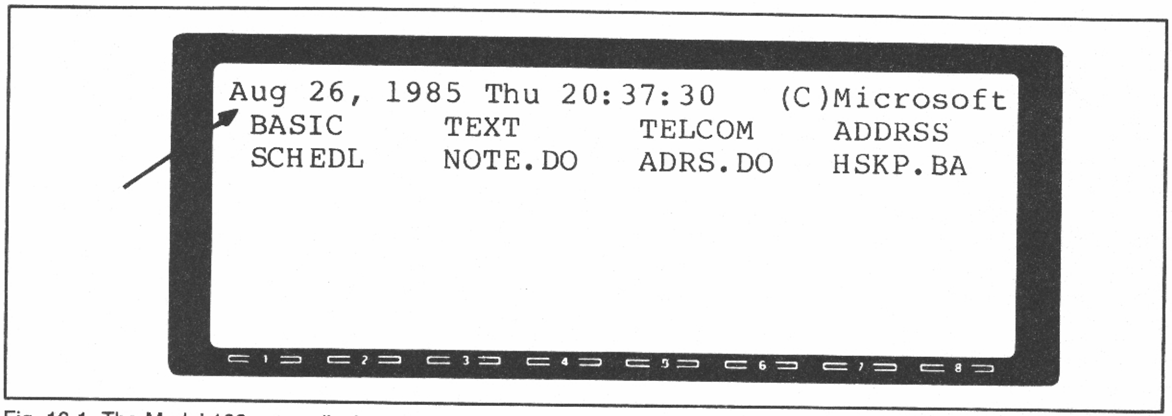
Time$ stores the time as a string variable in for format HH:MM:SS. The Model 100 uses the 24-hour clock. For example, BASIC stores 11:30 AM in time$ as 11:30:00 and stores 1:00 PM as 13:00:00.

Retrieving the Date or Time

Use date$ and time$ as constants in BASIC programs. For example, to print the current date (07/04/87), type the command:

```
1020 PRINT DATE$
```

To load the string variable TM$ with the current time (22:15:12), type the command:



```
Aug 26, 1985 Thu 20:37:30   (C)Microsoft
 BASIC      TEXT       TELCOM     ADDRSS
 SCHEDL     NOTE.DO    ADRS.DO    HSKP.BA
```

Fig. 16-1. The Model 100 menu displays the date, day, and time.

```
21100 TM$=TIME$
```

Setting the Date or Time

1. Invoke BASIC.
2. Type TIME$ = "HH:MM:SS" and press the enter key.
3. Type DATE$ = "MM/DD/YY" and press the enter key.

☐ You must type the date or time in quotes.

☐ For the date, you must type two digits for the month, two digits for the day, and two digits for the year. You must separate month, day and year by typing a slash (/).

☐ For the time, you must type two digits for the hour, two digits for the minute, and two digits for the seconds. You must separate hour, minute, and seconds by typing a colon.

For example, let's say it's 2:30 PM on July 4, 1985. To set the time and date, invoke BASIC, type the direct command,

```
DATE$ = "07/04/85"
```

press the enter key and type

```
TIME$ = "14:30:00"
```

and press the enter key.

POWER AND START-UP COMMANDS

BASIC provides you with commands to control the Model 100's built-in power and start-up functions. These commands are:

IPL
POWER
POWER, RESUME
MENU

For example, use the IPL command to run a program immediately upon start up. Use the power command to control the automatic power off feature and the menu command to exit BASIC and return to the menu.

Automatic Start-Up (IPL)

IPL is the Model 100's version of the TRSDOS

Auto command. You use IPL to tell the Model 100 to run a specific program each time you turn on the machine. You automatically run a BASIC program or to run one of the five built-in programs.

Using IPL. To run a program each time you turn on the model 100, the procedure is

1. Invoke BASIC.
2. Type IPL" (the program name and extension)", and press the enter key.

For example, let's say that each time you turn on the Model 100 you want to run your BASIC checkbook program, CHKBK.BA. You invoke BASIC, type the direct command

```
IPL "CHKBK.BA"
```

and press the enter key.

If you want the Model 100 to run the TELCOM program on start-up, type this command

```
IPL "TELCOM"
```

and press the enter key.

Cancelling IPL. If you have used IPL to instruct the Model 100 to run a program on power-up, you may want to cancel the instruction. The Model 100 manual doesn't tell you how to disable IPL, but it's pretty obvious. The steps are:

1. Invoke BASIC.
2. Type IPL by itself and press the enter key.

Power Commands

The Model 100 provides two power commands. One controls the automatic power-off feature; the other turns off the Model 100 from BASIC.

POWER (followed by a number or CONT) is the direct command you use to set the duration before automatic power off.

POWER OFF is the command you use in a BASIC program to turn off the Model 100. If you type a comma and "RESUME" after the command (POWER OFF, RESUME), BASIC resumes the program when you manually turn the Model 100 back on.

Controlling Automatic Power Off. To help you save on batteries, the Model 100 shuts itself off after a certain amount of time if no keys are pressed or if no program is running. BASIC's power command enables you to set the amount of time before automatic power off or to disable automatic power off. To use the power command

1. Invoke BASIC.
2. Type POWER.
3. Type a number from 10 to 255, or type CONT.

    - ☐ The number you type tells the Model 100 how long to wait before turning itself off. Each number, from 10 to 255 is equal to one-tenth of a minute. To set the waiting period at 10 minutes, type 100:POWER 100
    - ☐ Type CONT (for continuous) to disable the automatic power off. The Model 100 will not turn itself off.

4. Press the enter key.

For example, to set automatic power off for 20 minutes, type the command

```
POWER 200
```

and press the enter key.

To tell the Model 100 to stay on until you switch it off, type the command

```
POWER CONT
```

and press the enter key.

Power Off, Resume

You use the power off commnd in BASIC programs to turn off the Model 100 under certain conditions. For example, you can use the power off command in a program to turn off the Model 100 at midnight.

If you want Model 100 to resume the program when you turn it back on, type POWER OFF, and then a comma followed by the command resume.

To program the Model 100 to turn itself off.

1. In a program line, type POWER OFF.
For example, the line

```
250 IF TIME$ > "21:00:00" THEN POWER OFF
```

turns off the Model 100 if it's after 11:00 PM.

2. After typing POWER OFF, type a, (comma), and type RESUME to continue the program when you turn the Model 100 on again.

For example, if your program contained the lines

```
250   IF TIME$ > "21:00:00" THEN POWER
      OFF, RESUME
260   GOSUB 2000
```

line 250 would turn off the Model 100 at the first second past 9:00 PM. When you turn the machine back on, it would resume the program at line 260.

Power Off with TIME$. Actually, if you want the Model 100 to shut off at a certain time, you would probably use the power off command in conjunction with the ON TIME$ interrupt command. Interrupt commands such as ON TIME$ are discussed in Chapter 18.

Menu

To exit BASIC and return to the menu, you can press the F8 key or you can type the BASIC menu command. After pressing F8 *or* typing MENU and pressing the enter key, the Model 100 will exit BASIC and return you to the menu. An example of the direct command is

```
MENU
```

In a program, you would type something like

```
250 MENU
```

COMMANDS TO WORK WITH THE FUNCTION KEYS

The key and key list commands enable you to define and display the BASIC definitions of the eight function keys.

I really like this feature. When the Model 100 comes from the store, six of the function keys (F1 through F5 and F8) are already defined for BASIC. For example, F5 lists the current program; F8 exits BASIC and returns you to the menu. Two of the keys, F6 and F7, are undefined.

The key command enables you to set your own BASIC definitions for any of the eight function keys. The key list command lists the current definitions. You can, of course, press the label key to display the current definitions.

Listing Function Key Definitions. To list the current definitions of the function keys in BASIC, the procedure is

1. Invoke BASIC.

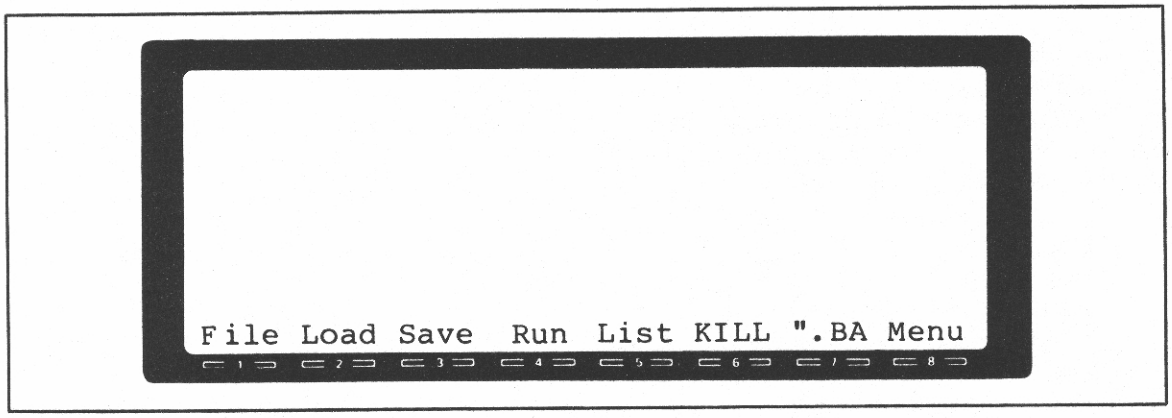


Fig. 16-2. When you define a function key, press the label key to display your definition. The function of F6 key and F7 key in the illustration has been set to kill or delete BASIC program files.

2. Type KEY LIST and press the enter key.

For example, if you type KEY LIST and press the enter key, BASIC lists the current definitions of the function keys (see Fig. 15-4).

Defining Function Keys. To define a function key, the procedure is

1. Invoke BASIC.
2. Type KEY. Then type the number of the key you want to define followed by a comma.
3. In quotation marks, type the definition you want and press the enter key.

For example, I find it useful to define F6 and F7 to "kill" BASIC program files I no longer need (Fig. 16-2). (In the following lines note that 34 is the ASCII code for a quotation mark and 13 is the ASCII code for ENTER.) To define F6 and F7, I type

```
KEY 6, "KILL" + CHR$(34)
KEY 7, ".BA" + CHR$(34) + CHR$(13)
```

Then I use F6 and F7 to easily delete BASIC program files from RAM. When I press F6 the Model 100 displays:

```
KILL "
```

I type the name of the program:

```
KILL"CHKBK
```

and press F7. The Model 100 displays:

```
KILL"CHKBK.BA"
```

and kills the file.

Restoring Function Key Definitions. To restore the original (default) function key definitions, the Model 100 provides a pair of machine-language subroutines you can call using the call command. The Model 100 Reference Manual gives these routines in the BASIC Keyword section of Part III. For your convenience I repeat them here.

```
CALL 23164,0,23366
CALL 27795
```

Remember to press the enter key after each line.

RAM FILE MANAGEMENT

Whether you use TEXT to type a document or

BASIC to write a program, the Model 100 stores documents and programs as RAM files. The menu lists all the files stored in RAM.

BASIC provides you with commands to manage these RAM files. To manage your RAM files, you use these BASIC commands:

FILES
KILL
NAME

Most often you will type these commands directly, but you can also use them in BASIC programs.

Files

When you display the menu, the Model 100 lists all the files stored in RAM (Fig. 16-3). However, you don't have to go to the menu to display your RAM files. Use the BASIC files command (or press F1) to display the files.

The files command is really helpful. For example, to kill a file, you don't have to go back to the menu to check the file name. The files command enables you to display all your RAM files without leaving BASIC. To do so

1. Press F1 or type **FILES** and press the enter key.

BASIC will display all the files stored in RAM. An example, to use it as a direct command, type

```
FILES
```

To use the command in a program, type

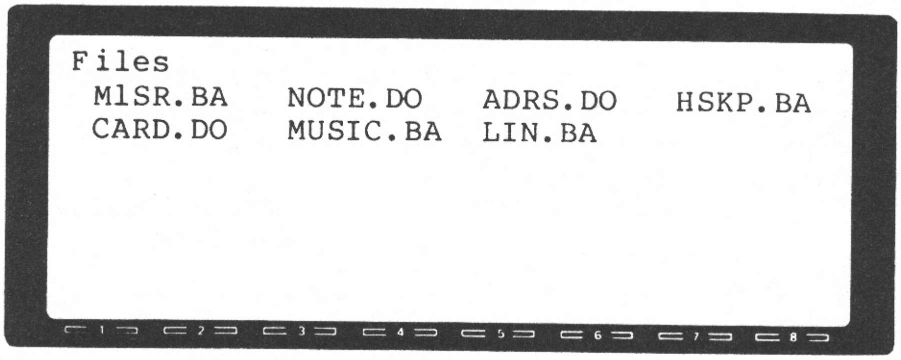
```
250 FILES
```

Kill

As you store documents and programs in RAM files, you may accumulate files you no longer need. What's more, you may run out of memory or use up all the available space in the menu, which has room for only 19 files. To delete files from RAM, you use BASIC's kill Command.

To kill a RAM file, the procedures is

1. Type **KILL**.
2. Type the name of the file in quotation marks, you must include the .DO or .BA extention to the file name.
3. Press the enter key.



```
Files
 M1SR.BA    NOTE.DO    ADRS.DO    HSKP.BA
 CARD.DO    MUSIC.BA   LIN.BA
```

Fig. 16-3. Press F1 key to display BASIC document and program files.

For example, to delete the SALES document, type the command

```
KILL "SALES.DO"
```

and press the enter key.

To delete the CHKBK program, type the command

```
KILL "CHKBK.BA"
```

and press the enter key.

You can also use the kill command in BASIC programs, as in the following sample program line.

```
2020  KILL "SALES.DO"
```

Name

The name command enables you to change the name of a RAM file. To use NAME

1. Type **NAME**.
2. Type the existing RAM file name in quotation marks.
3. Type **AS**.
4. Type the new file name in quotation marks.
5. Press the enter key.

To rename the document SALES.DO as MEMO.DO, for example, type

```
NAME "SALES.DO" AS "MEMO.DO"
```

and press the enter key.

To rename the program CHKBK.BA as BUDGET.BA, type

```
NAME "CHKBK.BA" AS "BUDGET.BA"
```

and press the enter key.

You can also use the name command in BASIC programs, as in the sample program line that follows.

```
3070  NAME "SALES.DO" AS "MEMO.DO"
```

MEMORY MANAGEMENT

In addition to the familiar BASIC commands such as poke, peek, and VARPTR, the Model 100 provides three memory commands that deserve special attention. They allow you to see how much *total* memory is available, to determine how much memory is available to BASIC, and to get the address of the "top" byte of memory available to BASIC. To manage memory, you use these BASIC commands (directly or in the programs):

MAXRAM
FRE
CLEAR
HIMEM

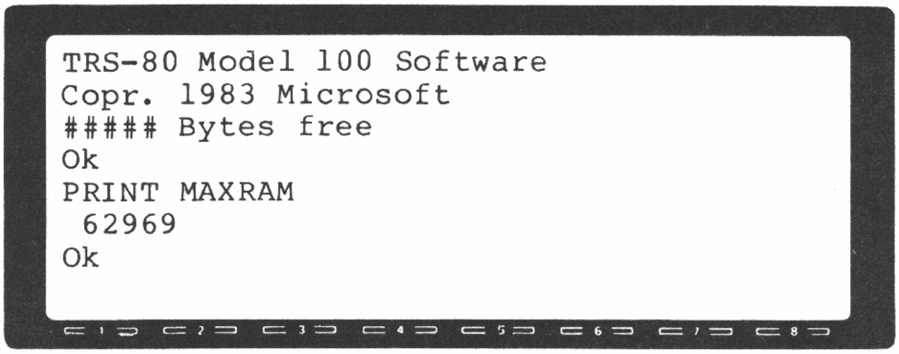
Maxram

MAXRAM is a constant. It contains the total number of bytes in your Model 100's RAM. You can use it to display the total RAM (Fig. 16-4). You can also use it in a program with the clear command to tell BASIC to use the total memory. (See the discussion on the clear command below.)

To display the total amount of memory, type **PRINT MAXRAM** and press the enter key.

Fre

FRE is really a function (most BASICs have a FRE function) that tells you how much memory is free, or currently unused. I single it out for mention because FRE is especially important on the Model 100. As you know, you use RAM memory to store



```
TRS-80 Model 100 Software
Copr. 1983 Microsoft
##### Bytes free
Ok
PRINT MAXRAM
 62969
Ok
```

Fig. 16-4. Use MAXRAM to display the maximum amount of memory free or available to you.

documents and programs in files. The more files you store, the less memory you will have available. FRE helps you keep track of available memory so that you don't run out of memory while you are running or editing a program.

Since FRE is a function, you must type a "dummy expression" in parentheses after it. A dummy *numeric* expression (such as 0) returns the number of available bytes for *number* storage. A dummy *string* expression such as ("") returns the amount of available bytes for *string* storage.

To display the amount of free (available) bytes of memory, for numbers, type **PRINT FRE(0)** and press the enter key (Fig. 16-5).

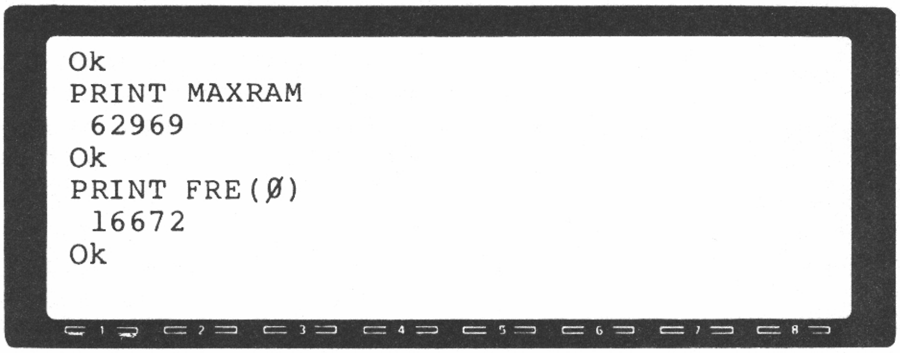
BASIC will display the number of bytes available for storing numbers.

To display the amount of bytes available for strings, type **PRINT FRE("")** and press the enter key.

BASIC will display the number of bytes available for storing strings.

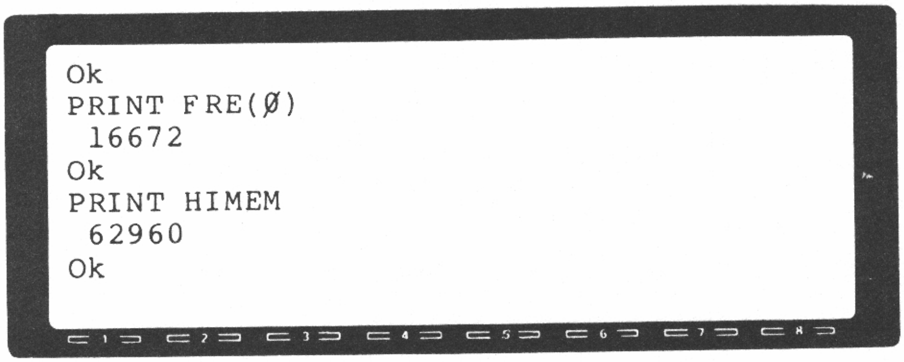
Clear

You are probably familiar with the clear com-



```
Ok
PRINT MAXRAM
 62969
Ok
PRINT FRE(0)
 16672
Ok
```

Fig. 16-5. The command PRINT FRE(0) will display the amount (in bytes) of memory available or free for number storage.



```
Ok
PRINT FRE(0)
 16672
Ok
PRINT HIMEM
 62960
Ok
```

Fig. 16-6. The command PRINT HIMEM will display the highest RAM address.

mand. However, the Model 100 version provides a capability that may be new to you. You use the clear command to set the highest memory address you want BASIC to use.

CLEAR clears all variables and closes any files that may be open. After typing **CLEAR**, you can specify the number of bytes you want to have available for string storage. After that, you can specify the address of high memory. For example

```
10   CLEAR
```

clears all variables and closes all files. If the form were

```
10   CLEAR 2000
```

it would clear all variables, close all files, and set aside 2000 bytes for string storage. The line

```
10   CLEAR 2000, 55000
```

clears all variables, closes all files, sets aside 2000 bytes for string storage, and sets the byte at address 55000 as the end of memory available to BASIC. If you have 62960 total bytes, that leaves byte 55001 through byte 62960 available for machine-language subroutines. Finally, the line

```
10   CLEAR 2000, MAXRAM
```

clears all variables, closes all files, sets aside 2000 bytes for string storage, and tells BASIC to use all available memory. In otherwords, no memory is reserved for machine-language routines or other special uses.

Himem

While FRE(0) tells you how many bytes are available, the HIMEM function tells you the *address* of the highest byte. For example, if you use the clear command to set a top address, using the HIMEM function will tell you where you set it.

To display BASIC's highest RAM address, type **PRINT HIMEM** and press the enter key (Fig. 16-6). You might use HIMEM, for example, to poke values above BASIC, as in the following lines.

```
 200   FOR I = HIMEM TO HIMEM + 20
 210   READ MC
 220   POKE I, MC
 230   NEXT I
1000   DATA 12,21,115,32,97 (etc.)
```

Chapter 17

Working with BASIC Programs

As I have said, Model 100 BASIC is fantastic. There is nothing "portable" about the programs you can write. The only drawback is the display; it is a little difficult to write and edit programs when you can view only eight lines at a time.

In this chapter I will discuss how you work with BASIC programs on the Model 100, including creating (writing) a new program and loading, listing, running, and editing existing programs. What's more, you'll learn how BASIC teams up with TEXT to form a powerful team.

CREATING NEW BASIC PROGRAMS

To create a new program on the Model 100, you write it and save it in RAM memory or on to cassette tape.

Writing a Program

To write a new BASIC program, the steps are:

1. Invoke BASIC from the menu.

The Model 100 will display Ok.

2. Type the program. Each line must begin with a line number, and after each line is typed you must press the enter key. For practice, use the embarrassingly simple program for finding square roots given in Fig. 17-1.

The program is now "resident" in BASIC. You can now save it, list it, edit it, and run it.

Saving New Programs

When you first write a program, you will almost always want to save it. You can do so in many ways including on cassette tape and through modem hookup with another computer. Usually, you save it in RAM. To save a program from BASIC to RAM, the procedure is as follows.

1. Press F3 or type SAVE.

The Model 100 will display SAVE "

```
10 ' Square root finder
20 INPUT "Type a whole number ";N
30 CLS
40 PRINT "The square root of ";N" is:"
50 PRINT SQR(N)
60 GOTO 20
```

Fig. 17-1. Program to calculate the square root of a number.

2. Type a name for the new program followed by a quotation mark.

You can use up to six characters for the program name. The first character cannot be a numeral. To save the program in Fig. 17-1, type

SAVE "ROOT"

3. Press the enter key.

The Model 100 will save the program in RAM as ROOT. BA (Fig. 17-2). It is not necessary to type the .BA extension to the program name. BASIC automatically adds it. Next time you display the menu, you will see your new program displayed at ROOT.BA.

Besides saving copy in RAM as a program, you can also save it in RAM as a *document*. In addition, you can save it on cassette tape, over a modem, over the RS-232, on a printer (same as LLIST), or to the screen (same as LIST), as shown in the following examples.

SAVE "ROOT",A

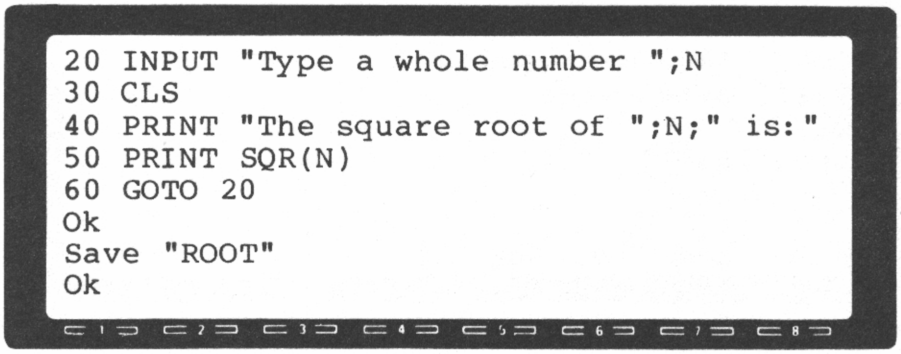
Saves the program in ASCII format as a document (ROOT.DO).

SAVE "CAS:ROOT"

Saves the program to a cassette recorder.

SAVE "MDM:7I1D"

Saves the program over the modem.



```
20 INPUT "Type a whole number ";N
30 CLS
40 PRINT "The square root of ";N;" is:"
50 PRINT SQR(N)
60 GOTO 20
Ok
Save "ROOT"
Ok
```

Fig. 17-2. Programs can be saved in RAM.

```
SAVE "COM:38N1E"
```

Saves the program over the RS-232

```
SAVE "LPT:"
```

Writes the program to the printer.

```
SAVE "LCD:"
```

Writes the program to the screen.

WORKING WITH BASIC PROGRAMS

To work with a program it must be "resident" in BASIC. You can either *type* a new program or *load* and *run* an existing program. Once the program is resident in BASIC, you can list it, edit it, and run it. (You can also run a BASIC program from the menu.)

Loading Programs

You can load a program into BASIC from RAM, cassette, or via the modem or RS-232.

To load a program from RAM to BASIC, i.e., to make the program "reside" in BASIC, the steps are as follows:

1. Invoke BASIC.
2. Press F2 or type `LOAD "`.
The Model 100 will display `LOAD "`.
3. Type the program name, followed by a quotation mark (Fig. 17-3) as in

```
LOAD "ROOT"
```

4. Press the enter key.

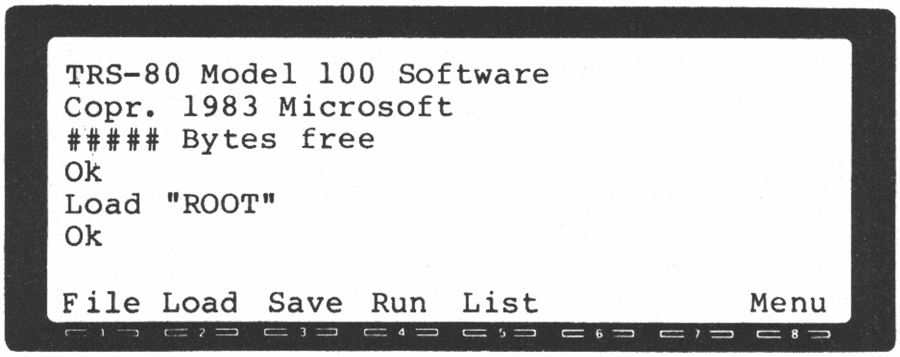
The Model 100 loads the program from RAM to BASIC. The program is now resident in BASIC.

Besides loading programs from RAM, you can also load and run programs, load programs from cassette, the modem, or the RS-232 using the following commands.

```
LOAD "CAS:ROOT"
```

Loads the program from cassette.

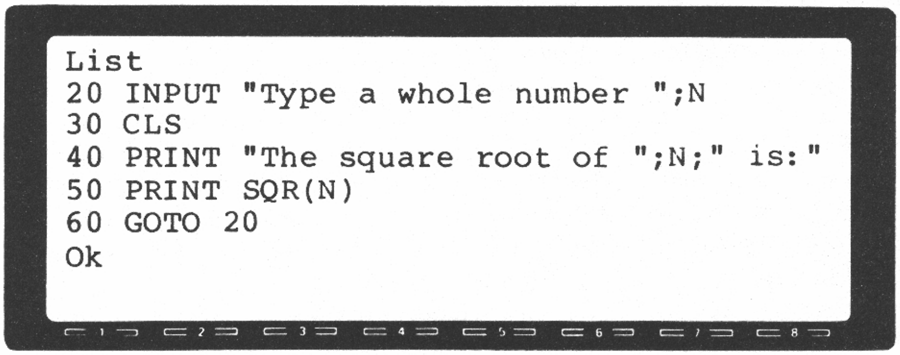
```
LOAD "MDM:7I1D"
```



```
TRS-80 Model 100 Software
Copr. 1983 Microsoft
##### Bytes free
Ok
Load "ROOT"
Ok

File Load Save Run List          Menu
```

Fig. 17-3. Loading a program.



```
List
20 INPUT "Type a whole number ";N
30 CLS
40 PRINT "The square root of ";N;" is:"
50 PRINT SQR(N)
60 GOTO 20
Ok
```

Fig. 17-4. Listing a program.

Loads the program from the modem.

```
LOAD "COM:38N1E"
```

Loads the program from the RS-232.

```
LOAD "ROOT",R
```

Loads and automatically runs the program after it finishes loading.

Once your program is resident (written or loaded) in BASIC, you are ready to list, run, and edit it. Here is where the Model 100 becomes a little peculiar. After you once save a program in RAM, you never save it again. If you load it and change it, the Model 100 automatically updates the RAM program file.

Listing a Resident Program

You can list the entire program, or a portion of the program.

Listing the Entire Program. To list a program, the steps are

1. Press F5 or type `LIST` and press the enter key.

The Model 100 will list the entire program from beginning to end (Fig. 17-4).

2. Press the break/pause key to halt the listing. Press the break/pause key again to resume the listing of the program.

Listing Part of a Program. To list a specific portion of the program, do the following:

1. Type `LIST`.

2. Type the first line number you want to list, type a hyphen, and type the last line number you want to list. For example

```
LIST 100 - 200
```

lists lines 100 through 200.

```
LIST 100 -
```

lists the program beginning at line 100.

```
List - 500
```

lists the program from the beginning through line 500.

3. Press the enter key.

You can use the break/pause key to halt and resume the listing.

Running Programs

After you have typed or loaded a program, you can run it. On the Model 100 you can run the resident program, run programs from RAM, cassette, the modem, or the RS-232.

Resident Programs. To run programs that are resident in BASIC, the procedure is as follows.

1. Press F4 or type `RUN` and press the enter key.

The Model 100 will run the resident program.

Programs from the Menu. To run a program listed in the menu, use this procedure.

1. Display the menu and position the cursor on the program. For example, if you loaded Fig. 17-1, position the cursor on `ROOT.BA`.
2. Press the enter key.

The Model 100 will automatically invoke BASIC and run the program.

Programs from BASIC. To run a program from BASIC, follow this procedure.

1. Invoke BASIC.
2. Type `RUN`.
3. Type the program name in quotation marks and press the enter key, as in the following.

```
RUN “ROOT”
```

Run Options. Besides running program from RAM, you can run programs from cassette, the modem, or the RS-232 using the commands in the forms listed as follows.

```
RUN “CAS:ROOT”
```

Runs the program from cassette.

```
RUN “MDM:7I1D”
```

Runs the program from the modem.

```
RUN “COM:38N1E”
```

Runs the program from the RS-232.

```
RUN “ROOT”,R
```

Leaves open any open files when running the new program.

USING TEXT WITH BASIC

One undisputed highlight of the Model 100 is its interactive use of BASIC and TEXT. You use TEXT to edit BASIC programs. (Chapter 20 shows you how to use BASIC to edit TEXT documents!)

Editing BASIC Programs

You can edit a program from BASIC or from TEXT, but before you dive in, make sure you know how to use the TEXT editing features explained in Chapter 6.

To edit a BASIC program from BASIC, the procedure is as follows.

1. Invoke BASIC.
2. Load the program you want to edit.
3. Type `EDIT`, and type the range of line numbers in one of the following forms.

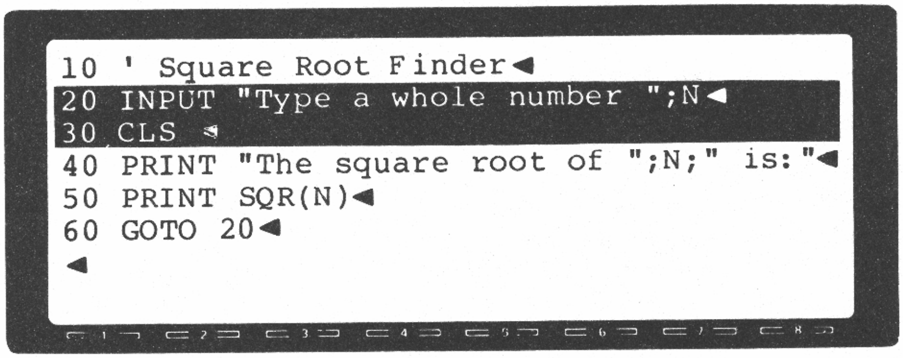
```
EDIT
```

Displays the entire program.

```
EDIT 100 - 200
```

1. Display program A and use select (F7) and cut (F6) to save the lines you want to cut and paste.

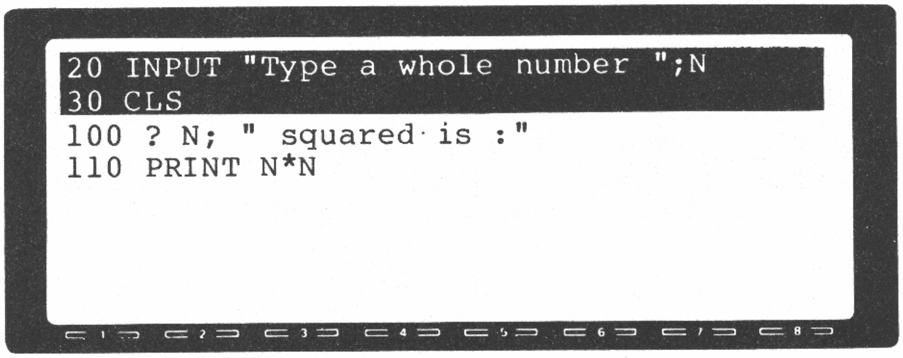
PROGRAM A



```
10 ' Square Root Finder◄
20 INPUT "Type a whole number ";N◄
30 CLS ◄
40 PRINT "The square root of ";N;" is:"◄
50 PRINT SQR(N)◄
60 GOTO 20◄
◄
```

2. Display Program B, position the cursor where you want the lines from Program A, and press the paste key. The lines selected and cut or copied from Program A will automatically be inserted in Program B.

PROGRAM B



```
20 INPUT "Type a whole number ";N
30 CLS
100 ? N; " squared is :"
110 PRINT N*N
```

Fig. 17-5. The Model 100's cut and paste features can be used for BASIC programs.

Displays lines 100 through 200.

```
EDIT 100 -
```

Displays the program beginning at line 100.

```
EDIT - 500
```

Displays the program from the beginning to line 500.

4. Press the enter key

The Model 100 will invoke the TEXT program and display the lines you asked to edit. The complete resources of TEXT are now at your disposal. Remember that you can always press the label key to display the TEXT function key definitions.

5. Use TEXT to edit the displayed lines. Move the cursor, then use insert, delete, select, cut, copy, paste, and so on.

6. When you have finished editing, press F8 to exit TEXT.

The model 100 will flash the message Wait as it changes the lines back into BASIC code.

Additional Ideas

TEXT and BASIC combine to form an extremely useful tool for working on BASIC programs.

Writing a Program. Invoke TEXT and write your program as a document. When you press F8 to exit your program, the Model 100 saves it as a document as ROOT.DO for example.

Then, invoke BASIC, and load the *documents* as in LOAD "ROOT.DO". BASIC converts the text to BASIC code as it loads. Save the program in RAM. The menu will list both the document and program version: ROOT.DO and ROOT.BA.

Turning a BASIC Program into a Document. To save a resident BASIC program as a document, type ,A after the program name to save it in ASCII format, as in the following.

```
SAVE "ROOT",A
```

In the menu you will see your program listed as a document, as in ROOT.DO. Now you can use TEXT to edit the program.

Cutting and Pasting Lines. You can use TEXT to cut lines from one program (Program A) and paste them in another program (Program B).

First, turn Program A (or part of Program A) into text using BASIC's edit feature, or save Program A as a document by typing SAVE "A",A.

Use the TEXT features select (F7), cut (F6), or copy (F5) to save the lines in the paste buffer.

Invoke BASIC, load Program B, and press the paste key. The Model 100 will insert the pasted lines into the correct position (Fig. 17-5).

For example, if you paste lines 100-200, the Model 100 inserts the pasted lines after 90 and before 210. If you paste a line 10, and if Program B already has a line 10, the pasted line replaces the original line.

Chapter 18

Outstanding Features of BASIC

The Model 100 provides all standard BASIC commands and functions—and then some. In this chapter I will discuss the features that make the Model 100's BASIC special. I have divided these special features into four categories.

Interrupt Commands
Sound
Display Control Commands
Open Command

INTERRUPT COMMANDS

The five interrupt commands branch to a subroutine under specific conditions. For example, if you want a program to tell you when it's ten o'clock, include the line:

```
150 ON TIME$ = "10:00:00 GOSUB 2000
```

At ten o'clock the program will go to the subroutine. When the subroutine is completed, the program will return to the point it left off before it interrupted itself.

One interrupt command—the on error command—is familiar. You use it to trap errors. We won't discuss that use here. The other four interrupts are:

ON COM (gosub if there is a communication from the RS-232).
ON KEY (gosub on one of the eight function keys)
ON MDM (gosub if there is communication from the modem)
ON TIME$ (gosub at a certain time)

You can also *enable, disable,* and *stop* the interrupts. In order to use an interrupt, you must first *enable* it. You can also disable the interrupt, or "stop" the interrupt. Stop means don't execute the

```
50 DD$ = MID$(DATE$,3,5)                    Get the day

60 DD = VAL(DD$)                            Its numeric value

70 IF DD/2=INT(DD/2) THEN TIME$ ON ELSE     If it's even
   TIME$ OFF                                (M,W,F) enable,
                                            else disable
80 ON TIME$ = "14:00:00" GOSUB 5000         At 2:00 remind
                                            me to call.
.
.
.
5000 FOR I = 1 TO 20                        Set off an alarm
5010 BEEEP                                  and remind me to
5020 NEXT I                                 call
5030 PRINT "Call the Home Office"
5040 RETURN
```

Fig. 18-1. A simple interrupt program.

interrupt now, but do it later if the interrupt condition occurred. For example, **TIME$ ON** enables the ON TIME$ interrupt, **TIME$ OFF** disables it; **TIME$ STOP** prevents the interrupt but executes it later if you "reenable" it, and if the condition occurred. Don't puzzle too long over this sentence, the example that follows will clear up any confusion.

Let's use ON TIME$ as an example. You are an executive, and you keep the Model 100 on your desk. You type appointments in your NOTE.DO file using the (SCHEDL) program as you make them. You also delete appointments and update them. So, you write a BASIC program called NOTWRT.BA (for Note Write) that enables you to type, edit, and delete appointments. Each day you keep the program running.

On Monday, Wednesday, and Friday you must call the home office at 2:00 PM to report on the latest sales figures. So in your NOTWRT.BA program you include the lines shown in Fig. 18-1.

Monday is the eighth day of the month; Wednesday, the tenth; Friday, the twelfth. If the day of the week is an even number, line 70 enables the interrupt; otherwise, line 70 disables it.

On Monday, Wednesday, and Friday at 2:00 PM the program will branch to the subroutine at line 5000 and beep to remind you to call the home office. The program will then resume whatever it was doing at the time it was interrupted.

You usually stop an interrupt when you include more than one interrupt in a program.

Let's add a new wrinkle to our executive's program (Fig. 18-2). The executive's Model 100 is hooked up by modem to field offices. At least once a day, sales figures come in from the field offices, so he includes the ON MDM (modem) interrupt to pause the program and record the sales figures when they come in.

What if the sales figures come in just before two o'clock? He doesn't want to interrupt the modem interrupt with the TIME$ interrupt. That's why you use the STOP command. When the modem interrupts the program in line 40 of Fig. 18-2, the Model 100 branches to a subroutine at line 4000. Line 4010 stops the time interrupt. Line 4500 reenables the time interrupt before it returns.

In our example, if the modem interrupt starts at 1:55 PM and stops at 2:10 PM, the program temporarily stops the time interrupt but remembers that it tried to happen. When the modem subroutine reenables the time interrupt, the program executes it.

Now that I have explained the basic idea, let me briefly describe each of the four interrupt commands.

On Time$

Use the ON TIME$ command to branch to a subroutine at a specific time, or before or after a specific time.

```
"An Interrupt Program That Demonstrates STOP."

30 MDM ON                                          Enable modem interrupt.

40 ON MDM GOSUB 4000                               If the modem is called, store
                                                   incoming data.

50 DD$ = MID$(DATE$,3,5)                           Get the day.

60 DD = VAL(DD$)                                   Numeric value of the day.

70 IF DD/2=INT(DD/2) THEN TIME$ ON ELSE TIME$ OFF  If it's an even-numbered day _
                                                   (M,W,F), enable; else disable.

80 ON TIME$ = "14:00:00" GOSUB 5000                At 2:00, remind me to call.

.
.
.
4000 'Store data from Modem

4010 TIME$ STOP                                    Temporarily stop time interrupt.

(4020 - 4400 Modem Subroutine)

4500 TIME$ ON                                      Reenable time interrupt.

4510 RETURN
.
.
.
5000 FOR I = 1 TO 20                               Set off an alarm,

5010 BEEEP
5020 NEXT I

5030 PRINT "Call the Home Office"
5040 RETURN                                        and remind me to call.
```

Fig. 18-2. An interrupt program that demonstrates the use of the stop command.

Enabling, disabling, and stopping the interrupt are accomplished with the following commands.

TIME$ ON

Enables the interrupt.

TIME$ OFF

Disables the interrupt.

TIME$ STOP,

Disables the interrupt, but remembers if the necessary condition arose. The program begins execution again at the point of the interrupt when it is reenabled with TIME$ ON.

The form to use is ON TIME$ = "HH: MM:SS:" GOSUB line number as in the following two examples

```
30   TIME$ ON
40   ON TIME$ = "13:15:00" GOSUB 4000
```

```
30   TIME$ ON
40   ON TIME$ > "10:00:00" GOSUB 4000
```

On Key

Use the ON KEY command to branch to a subroutine when you press one of the eight function keys.

Enabling, disabling, and stopping the interrupt are accomplished with the following commands.

KEY ON

Enables the interrupt.

KEY OFF

Disables the interrupt.

KEY STOP,

Disables the interrupt, but remembers if the necessary condition arose. The machine begins execution of the program again at the point of an interrupt reenabled with KEY ON.

The form to use is ON KEY GOSUB list of line numbers as in the following example.

```
30   KEY ON
40   ON KEY GOSUB 1000, 2000, 3000, 4000
```

If you press F1 the program will branch to line 1000. If you press F2 the program will branch to line 2000. If you press F3, it will branch to line 3000, and so on.

On MDM

Use the ON MDM command to branch to a subroutine when communication comes in through the modem.

Enabling, disabling, and stopping the interrupt are accomplished with the following commands.

MDM ON

Enables the interrupt.

MDM OFF

Disables the interrupt.

MDM STOP,

Disables the interrupt, but remembers if the necessary condition arose. The Model 100 begins execution of the program again at the point of the interrupt when it is enabled with MDM ON.

The form to us is ON MDM GOSUB and a line number as in the following example.

```
30   MDM ON
40   ON MDM GOSUB 8000
```

On COM

Use ON COM to branch to a subroutine when communication comes in through the RS-232.

Enabling, disabling, and stopping the interrupt are accomplished with the following commands.

```
COM ON
```

Enables the interrupt.

```
COM OFF
```

Disables the interrupt.

```
COM STOP,
```

Disables the interrupt, but remembers if the necessary condition arose. Program execution begins again at the point of an interrupt when reenabled with COM ON.

The form to use is ON COM GOSUB and a line number as in the following.

```
30  COM ON
40  ON COM GOSUB 8000
```

SOUND

BASIC offers more than the usual display capabilities. The Model 100 includes a piezoelectric sound generator that you control with BASIC.

BASIC provides four commands for creating sound. You use the beep and sound commands to make noise; you use sound on and sound off commands for noise control with the cassette or modem.

Beep

The beep command makes the Model 100, well, beep (for about half a second). It's easy to use. To beep, just type BEEP.

The following line makes the Model 100 beep twenty times.

```
4000 FOR I = 1 to 20: BEEP: NEXT
```

Sound Command

The sound command can be music to your ears. It enables you to specify both the pitch and duration of a sound.

Type SOUND followed by a *pitch* value, a comma, and a *duration* value. For example

```
SOUND 9394,100
```

A chart in User's Manual lists the numeric value for each note. The Model 100's range is five octaves (each from the musical note G through F#), and, if you are into 12-tone scales, everything in between. For example, to sound a musical C note, you type the pitch value 9394.

You also define the duration of the tone by typing a number from 0 to 255. As the manual says, divide the number by 50 to find the approximate length in seconds: 50 is about one second, 100 is two seconds, and so on.

This line sings a C note for two seconds:

```
100  SOUND 9394,100
```

Sound On and Off

When you load a file from cassette or establish communications over the modem, the Model 100 makes, in my opinion, a shrill and unpleasant noise. What's happening is that the Model 100 plays you the "bit stream" as it comes over the cassette interface or modem. Thankfully, in their benign wisdom, Tandy provides a command that enables you to silence this shrill noise.

Type SOUND OFF to turn off the bit stream.

Type SOUND ON if for some unknown reason you want to hear the bit stream.

DISPLAY CONTROL

All the usual commands to control the display are available on the Model 100. They are the print command, print @ command, print tab command, and print using command. We won't go into the garden variety commands here, but there are a few special commands that command our interest, the most interesting is the line command.

You can control the display by characters or pixels.

Characters

The Model 100 displays 8 lines of 40 characters each. Thus, there are 320 character positions (8 × 40) on the screen. Since the first space is designated as 0, there are 0 through 319 character positions (Fig. 18-3).

For example, the command

```
PRINT @ 140, "X"
```

prints an X at character position 140.

Pixels

Each character position is really a "box" on the screen made of 48 smaller boxes. Each box is called a *pixel* (Fig. 18-4). So, looking at the display in pixels, each line is 240 pixels long, and there are 64 lines of pixels (Fig. 18-5).

Model 100 BASIC provides three commands that enable you to work with pixels: PSET, PRESET, LINE.

PSET and PRESET. To turn on a pixel (darken it), you type the command PSET followed by (in parentheses) the x coordinate and the y coordinate of the pixel you want to turn on. For example, to turn on the pixel in the middle of the screen, you type

```
PSET (119,31)
```

The Model 100 will turn on the 119th pixel in the 31st line (Fig. 18-6).

To turn off a pixel (make it disappear), you type the command PRESET followed by (in parentheses) the x coordinate and the y coordinate of the pixel you want to turn off. For example, to turn off the pixel in the middle of the screen, you type

```
PRESET (119,31)
```

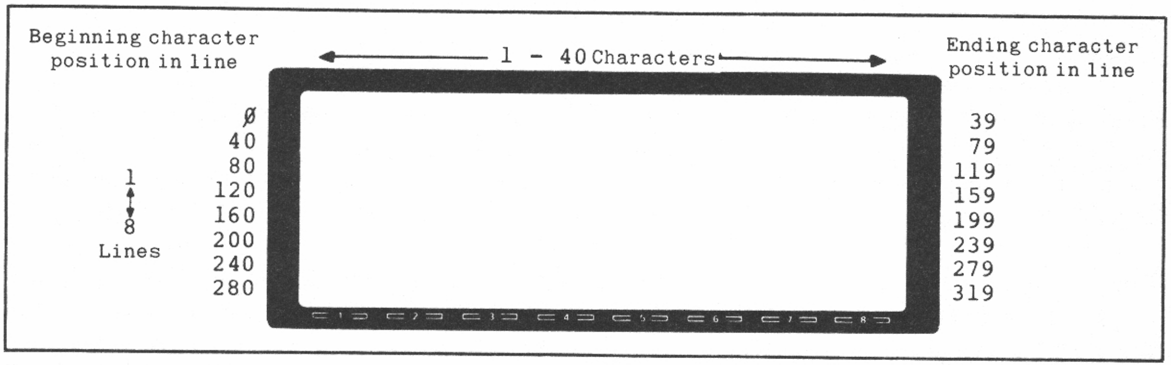


Fig. 18-3. The screen character positions.

Line. The line command is great; it enables you to

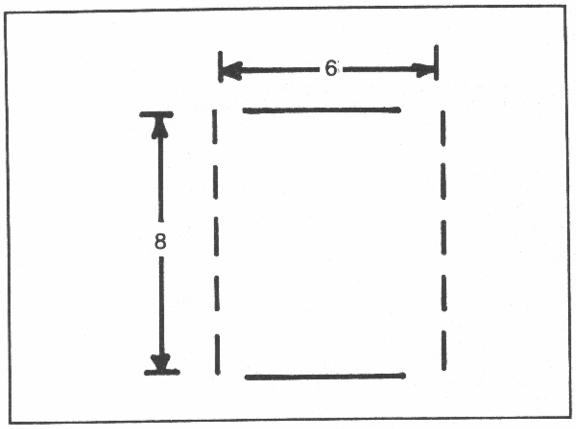


Fig. 18-4. A single character is made up of 48 points.

draw (or erase) a line or box between two points (pixels). What's more, it enables you to fill in the box. It's great for making graphs.

To use the line command, you type **LINE** followed by (in parentheses) the x and y coordinates for the first point, followed by (in parentheses) the x and y coordinates for the second point. The Model 100 draws a line between the two points.

To draw a line in the middle of the screen—from point 60, 32 to point 180, 32—you type

```
LINE (60,32) - (60,180)
```

as shown in Fig. 18-7.

Switch. The optional switch numeric expression adds more dimension to the line command. If you type nothing or an odd value (such as 1) after the two coordinates, the Model 100 draws a line. If you type an even value (such as 1) after the two coordinates, the Model 100 erases the line. For example, to erase the line in Fig. 18-7, you would type

```
LINE (60,32) - (180,32), 0
```

Box. If after the switch command, you type B, the Model 100 draws a box between the two sets of coordinates. Let's say you want to draw a box with corners at point 60,16 and point 48,180. You type

```
LINE (60,16) - (180,48), 1,B
```

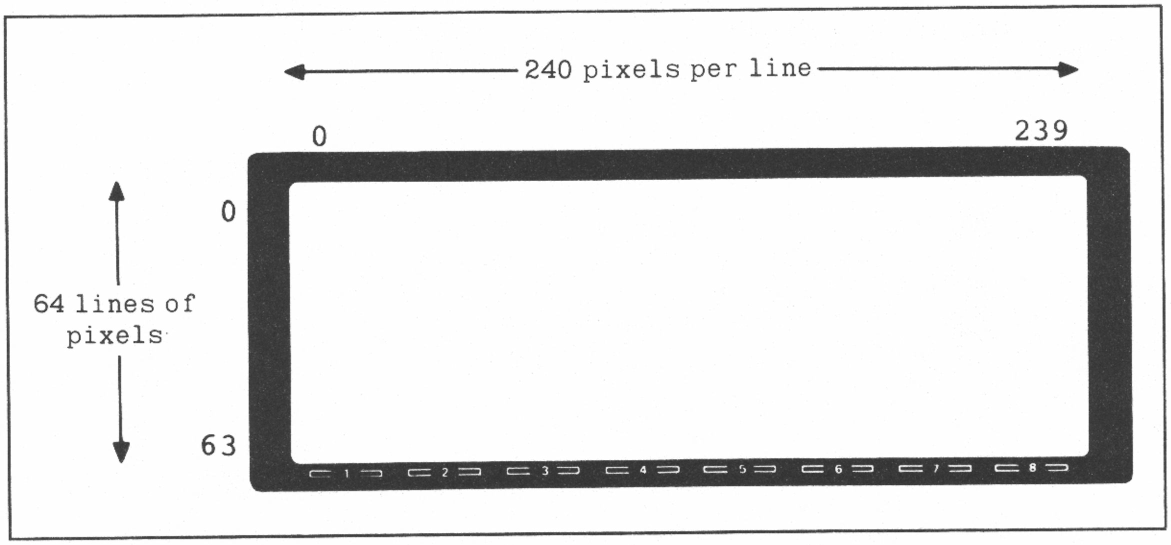


Fig. 18-5. The screen pixel positions.

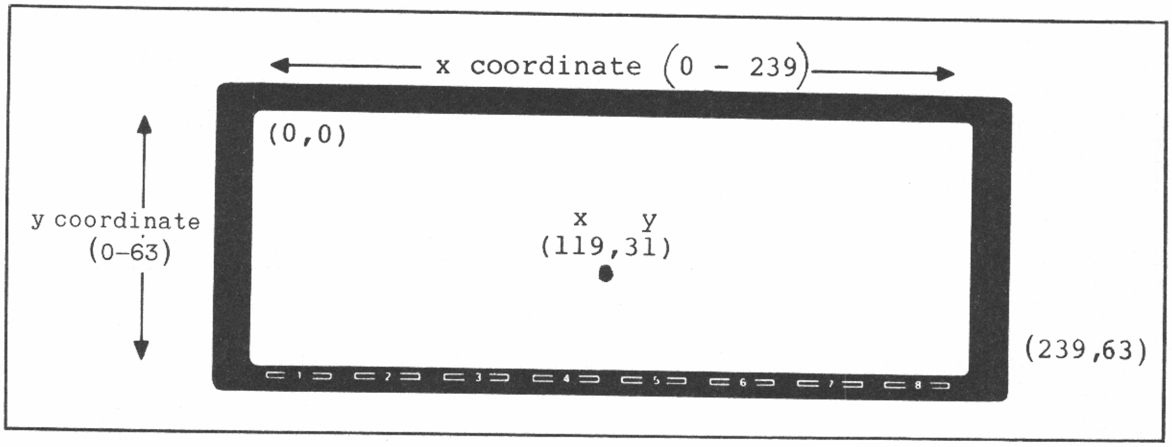


Fig. 18-6. The point drawn is at x coordinate 119 and y coordinate 31.

as illustrated in Fig. 18-8.

Box Fill. If after the switch command, you type BF (for "box fill"), the Model 100 draws a box between two points and fills it in to make a solid block. Let's say you want to draw the box in Fig. 18-8, but you want it filled in. You type

```
LINE (60,16) - (180,48), 1,BF
```

as shown in Fig. 18-9.

THE OPEN COMMAND

The open command is BASIC's doorway to the other programs. It is what makes BASIC truly interactive. For example typing, OPEN RAM enables BASIC to open a TEXT document, then edit and rewrite it. OPEN MDM (for modem) enables you to write programs that input and output through

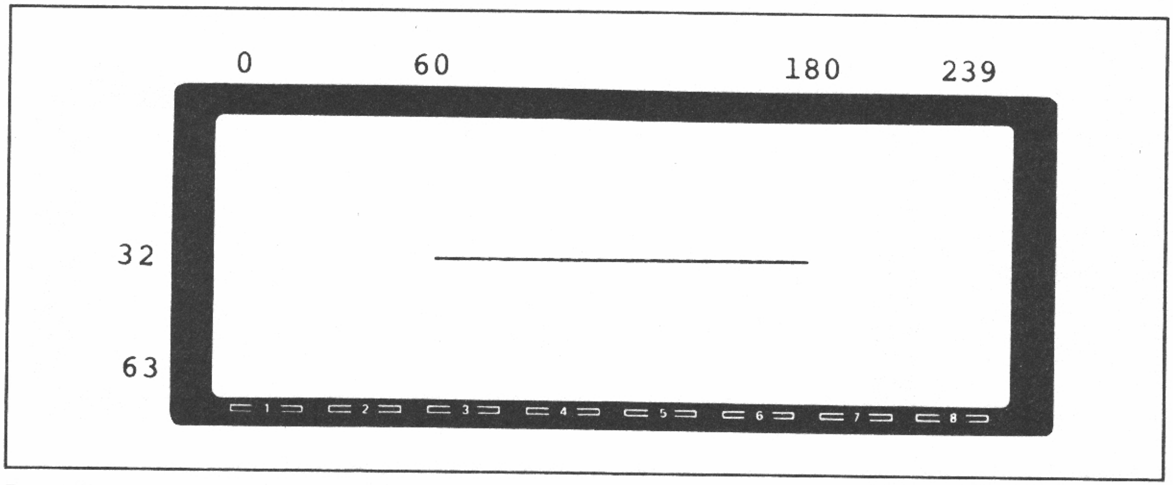


Fig. 18-7. Using the line command, you can draw this line beginning at point 60,32 to point 180,32.

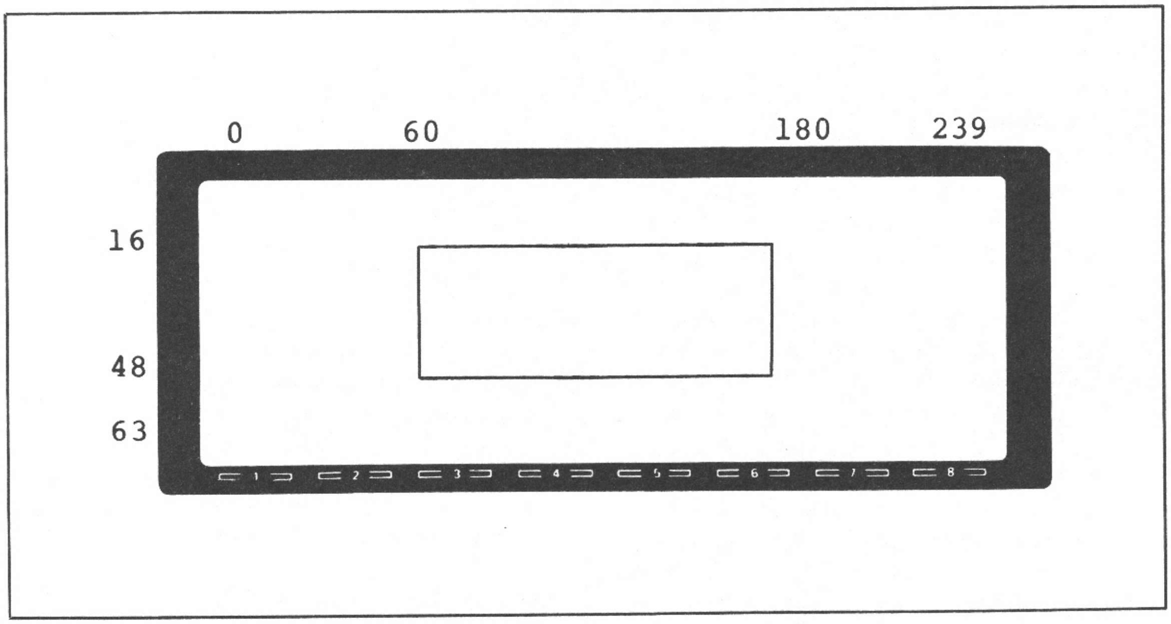


Fig. 18-8. The command line (60,16)-(180,48),1,B draws this box.

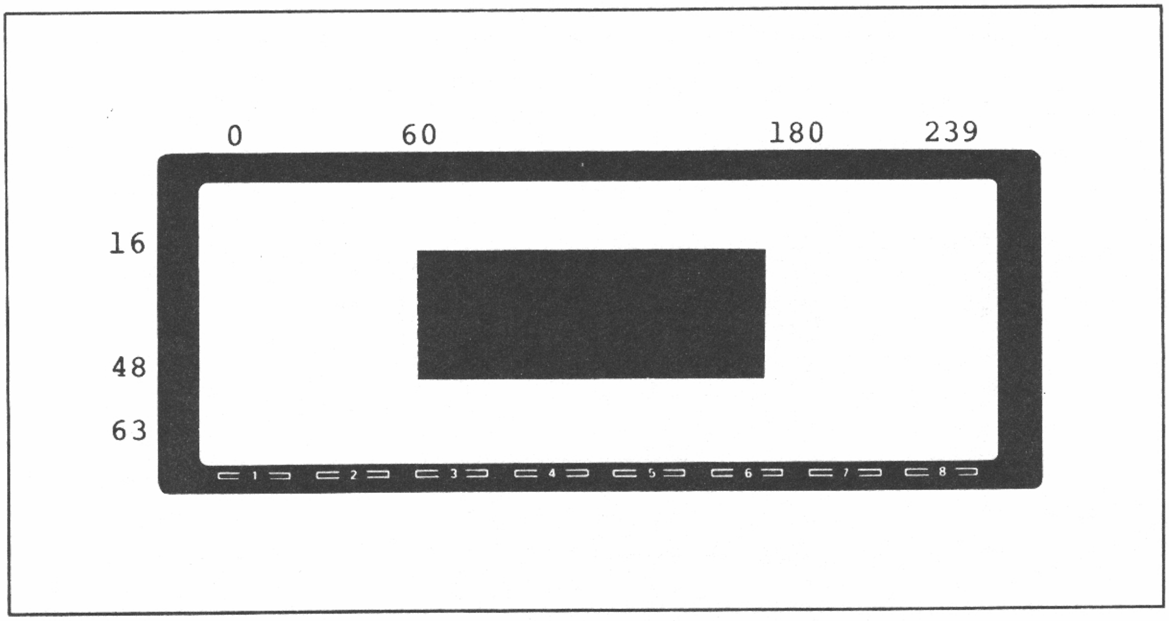


Fig. 18-9. Ending a line statement with BF fills in the box.

the Modem. OPEN COM enables you to control the RS-232 from BASIC.

Six Kinds of Files

To really take advantage of the Model 100, it's helpful to know that this portable really expands the definition of *file*. The Model 100 enables you to treat RAM, the RS-232 interface, and the modem as "files" for input, and append. You can treat the cassette interface as a file for input or output. You can also treat the printer and screen as files for output (Fig. 18-10).

Perhaps you are used to working with disk or cassette files: you open a file, output (write) data to the file, input (read) data from the file, and so on. When you are finished, you close the file. As you know, when you open one or more files, the computer sets aside some memory as a buffer, a temporary holding space for the data that you are inputting and outputting. The Model 100 works the same way.

File I/O

To use any device as a file, there are three main commands:

MAXFILES
OPEN
CLOSE

You use the maxfiles command to specify the total number of files you will use in your program. You can use the open command to open one or more files for input, output, or append, and to assign each file a specific number. You use PRINT# statements to output (write) to an open file. You use INPUT# statements to input (read) from an open file. When you are finished outputting or inputting, you use the close command to close the file.

For example, let's say you have saved the BASIC program CHKB in RAM as TEXT. (See Saving Programs in Chapter 17). The little program in Fig. 18-11 opens the TEXT document and stores

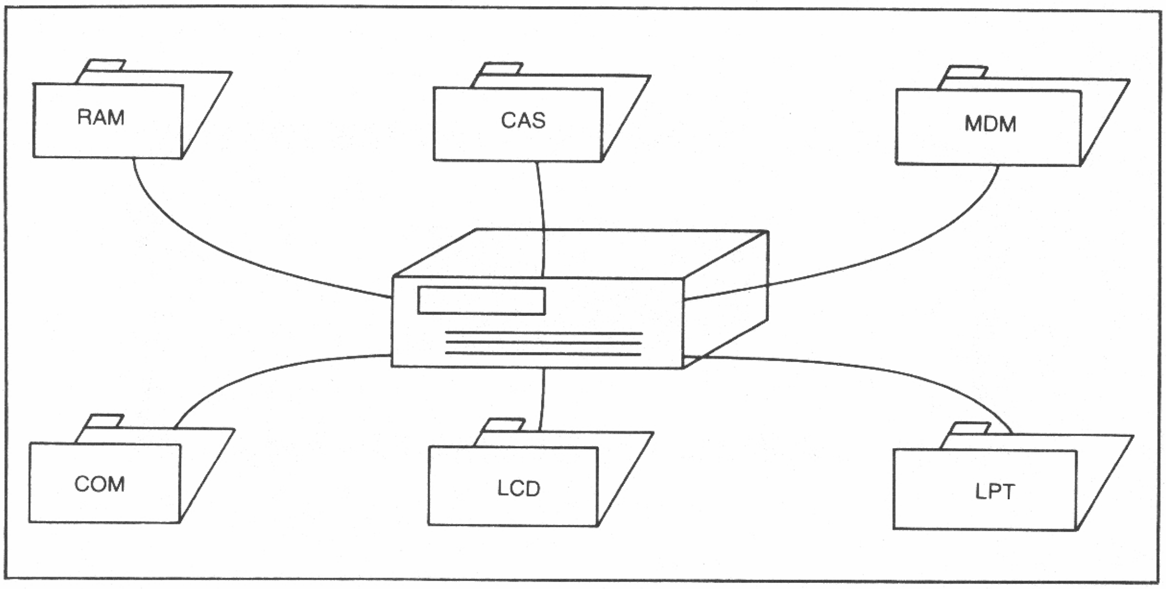


Fig. 18-10. You use the open command to open six kinds of files.

```
10 CLEAR 1000
20 MAXFILES = 1
30 DIM LN$(80)
100 OPEN "CHKBK.DO" FOR INPUT AS 1
110 FOR I = 1 TO 80
120 IF EOF(1) THEN GOTO 200
130 LINE INPUT#1, LN$(I)
140 NEXT I
200 CLOSE
```

Fig. 18-11. These lines open a document and read lines into LN$.

each line in the dimensioned LN$ variable. You might use these lines as part of a program that edits or renumbers BASIC programs.

The three major file I/O commands are: MAXFILES, OPEN, and (discussed previously) CLOSE.

MAXFILES. MAXFILES is a constant that contains the maximum number of files BASIC can use. When you start up, BASIC defaults to file 1. You can set MAXFILES to any number between 0 and 15.

To display the maximum number of available files, type

```
PRINT MAXFILES
```

To set the maximum number of available files, type

```
MAXFILES = #
```

For example, the following line means that BASIC can work with up to three files.

```
20   MAXFILES = 3
```

OPEN. You use OPEN to open files and to specify the following.

☐ Kind of file (RAM, MDM, etc.)
☐ Name of the file (or for the modem or RS-232, the "communication parameters")
☐ Action (input, output, or append)
☐ Number of the file

The format for the open command is always the same. Type **OPEN** followed by the kind of file, the name or communication parameters, **FOR** and the action (input, output, append), **AS** and a number from 1 to 15 (Fig. 18-11):

You can open six kinds of files: RAM, cassette, RS-232, modem, display, or printer.

RAM files are the files you have stored in the Model 100's memory, the ones displayed in the menu. You can open RAM files (such as BASIC programs or TEXT documents) for input, output, or append. For example

```
OPEN "RAM:CHKBK.BA" FOR INPUT AS 1
```

as shown in Fig. 18-12.

If you open a RAM file for output that does not yet exist, BASIC creates one.

If you type nothing before the file name, BASIC assumes you are opening a RAM file:

```
OPEN "CHKBBK" FOR INPUT AS 1
```

You can open cassette files (such as BASIC programs or TEXT documents) for input or output. For example

```
OPEN “CAS:GAME” FOR OUTPUT AS 1
```

If you are opening a cassette file for output that does not yet exist, BASIC creates one.

OPEN enables you to output or input from both the modem and the RS-232. When you open the modem or RS-232 as a file, you type the communications parameters instead of a file name. You type the parameters in the same format you use with STAT in the TELCOM program: baud rate, word length, parity, stop bits, line status. (If you are opening the modem, you leave out the baud rate. MDM sets it automatically at 300.)

For example, to open the modem for CompuServe input, type:

```
OPEN “MDM:7I1D” FOR INPUT AS 1
```

To open the RS-232 for output to the Model III, type:

```
OPEN “COM:38N1E” FOR OUTPUT AS 1
```

Depending on the kind of file, you select one of three actions when opening a file: input, output, or append.

Specify input to read data from the open file; specify output to write data (starting at the beginning of the file); and specify append to write data starting at the end of the file.

If you open RAM, the modem, or the RS-232 (COM), you can choose any of the three actions.

If you open the cassette (CAS), you can choose input or output, but you cannot choose append.

If you open the printer (LPT) or screen (LCD), you can choose output only.

File Number (FOR #)

You assign each file its own number. The real advantage of numbering open files is that the same program lines can do the same job on different files. For example, these lines write program lines to file number 3:

```
500   FOR I = 1 TO 100
510   PRINT#3, LN$(I)
520   NEXT I
```

Depending on the OPEN statement, you can use these same lines to output to RAM, the cassette, the modem, the RS-232, the screen, or the printer, as shown in the following examples.

```
OPEN “RAM:CHKBK.DO” FOR OUTPUT AS 3
```

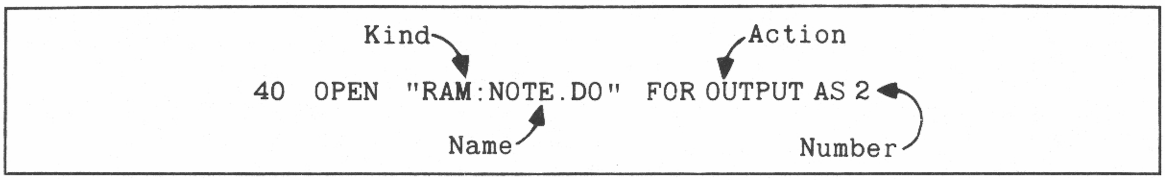
Outputs to RAM.

```
OPEN “CAS:CHKBK.DO” FOR OUTPUT AS 3
```

Outputs to the cassette.

```
OPEN “MDM:7I1D” FOR OUTPUT AS 3
```

Outputs to the modem.



```
40  OPEN  "RAM:NOTE.DO"  FOR OUTPUT AS 2
```

Fig. 18-12. The parts of the open command.

```
OPEN “COM:38N1E” FOR OUTPUT AS 3
```

Outputs to the RS-232.

```
OPEN “LCD:” FOR OUTPUT AS 3
```

Outputs to the screen.

```
OPEN “LPT:” FOR OUTPUT AS 3
```

Outputs to the printer.

Chapter 19

Demonstration Programs

Here are some programs that demonstrate the commands discussed in Part 5. Some of the programs are useful; others simply enable you to try out such commands as LINE, SOUND, and ON TIME$.

GOOD HOUSEKEEPING PROGRAM

The program in Fig. 19-1 performs certain housekeeping chores such as killing a file, (re)naming a file, and turning off the power.

1. Type the program in Fig. 19-1.
2. Save the program as HSKP (for housekeeping).
3. Run the program and edit out any bugs.
4. Use IPL so that the Model 100 runs the HSKP each time you turn it on.
5. Define the F6 function key to run HSKP from BASIC. To do so, type

```
KEY 6, "RUN " + CHR$(34) + "HSKP" + CHR$
                (34) + CHR$(13)
```

in which 34 is the ASCII code for a quotation mark and 13 is the ASCII code for ENTER. (Remember, to restore the original function key definitions, use the call commands.)

LINE DEMO PROGRAM

The program in Fig. 19-2 enables you to experiment with the different possibilities of BASIC's line command.

1. Type the program in Fig. 19-2.
2. Save the program as LN (for line).
3. Run it, edit out any bugs, and try typing different values.

A SOUND PROGRAM

The program in Fig. 19-3 enables you to experiment with BASIC's sound command. It turns the bottom row of keys into a piano.

1. Type the program in Fig. 193.
2. Save the program as MUSIC.
3. Run the program, edit out any bugs.
4. Try playing a few tunes, using the bottom row of keyboard keys to play a tune.

Be sure that the caps/lock key is depressed; or write the program so that it accepts either upper- or lowercase letters.

ALARM CLOCK PROGRAM

The program in Fig. 19-4 enables you to experiment with BASIC's ON TIME$, ON KEY, and beep commands. The program turns the Model 100 into an alarm clock with a "snooze" alarm.

1. Type the program in Fig. 19-4.
2. Save the program as ALARM.
3. Run the program.

Fig. 19-1. Good Housekeeping program (continued on page 136).

```
 100 'A PROGRAM FOR MODEL 100 HOUSEKEEPING
 120 CLEAR 500, MAXRAM
 130 DEFINT A-Z
 200 'MENU
 210 CLS:PRINT TAB(10) "*** HOUSEKEEPING ***"
 220 PRINT TAB(5) "1...Display the time."
 230 PRINT TAB(5) "2...Display the date."
 240 PRINT TAB(5) "3...Display the Function Keys."
 250 PRINT TAB(5) "4...Kill a file."
 260 PRINT TAB(5) "5...Name a file."
 270 PRINT TAB(5) "6...Return to menu."
 280 PRINT TAB(5) "7...Power off.  ";
 290 GOSUB 8000
 300 ON VAL(IR$) GOSUB 1000,2000,3000,4000,5000,6000,7000
 310 GOTO 200
1000 'DISPLAY THE TIME
1010 CLS
1020 PRINT "The time is ";TIME$
1030 PRINT: PRINT "Press any key to continue."
1040 GOSUB 8000
1050 RETURN
2000 'DISPLAY THE DATE
2010 CLS
2020 PRINT "Today's date is ";DATE$
2030 PRINT:PRINT "Press any key to continue."
2040 GOSUB 8000
2050 RETURN
3000 ' DISPLAY THE FUNCTION KEYS
3010 CLS
3020 PRINT "Here are the current function key definitions:"
3030 KEY LIST
3040 PRINT:PRINT "Press any key to continue."
3050 GOSUB 8000
```

```
3060 RETURN
4000 'KILL A FILE
4010 CLS
4070 FILES
4080 PRINT "Which file do you want to delete?"
4090 GOSUB 9000
4100 KF$ = FL$
4110 CLS: PRINT "Do you really want to kill ";KF$;" (Y/N)?"
4120 GOSUB 8000
4130 IF IR$ <> "Y" THEN RETURN
4140 KILL KF$
4150 RUN"HSKP"
5000 ' RENAME A FILE
5010 CLS
5030 FILES
5040 PRINT "Which file do you want to rename?"
5050 GOSUB 9000
5060 PN$ = FL$
5070 CLS
5080 PRINT "Type the new name for ";PN$;":"
5090 GOSUB 9000
5100 NN$ = FL$
5110 CLS
5120 PRINT "Do you really want to change ";PN$
5125 PRINT "to ";NN$;" (Y/N)?"
5130 GOSUB 8000
5140 IF IR$ <> "Y" THEN RETURN
5150 NAME PN$ AS NN$
5160 RETURN
6000 ' RETURN TO MENU
6010 MENU
7000 'POWER OFF
7010 POWER OFF,RESUME
7020 RETURN
8000 'INKEY$ SUBROUTINE
8010 IR$=INKEY$: IF IR$ = "" THEN GOTO 8010
8030 C = ASC(IR$): IF C > 90 THEN C = C-32: IR$=CHR$(C)
8040 RETURN
9000 'FILE NAME INPUT SUBROUTINE
9010 LINE INPUT FL$
9020 PRINT "Is ";FL$;" a BASIC file or a DOCUMENT (B/D)?"
9030 GOSUB 8000
9040 IF IR$ = "B" THEN EX$=".BA" ELSE EX$ = ".DO"
9045 IF LEN(FL$) < 3 THEN FL$ = FL$ + EX$
9050 IF MID$(FL$,LEN(FL$)-2,1) <> "." THEN FL$ = FL$ + EX$
9060 RETURN
```

Fig. 19-2. Line demonstration program (continued on page. 138).

```
 100 ' A PROGRAM THAT LETS YOU TRY OUT
 110 ' LINE
 120 ' WITHOUT RETYPING THE COMAND EACH TIME
 200 DEFINT A-Z: BC=0
 210 CLS
 220 PRINT "Set the four X Y coordinates."
 230 GOSUB 1000
 240 PRINT "Set SW (0 for even, 1 for odd)"
 250 GOSUB 1100
 260 PRINT "Set B"
 270 GOSUB 1200
 280 CLS: GOTO 390
 300 PRINT "Press any key to continue.": GOSUB 3000
 310 CLS
 320 PRINT "Try another combination."
 330 PRINT: PRINT "1....Change coordinates"
 340 PRINT "2....Change SW"
 350 PRINT "3....Change B"
 360 GOSUB 3000
 370 IR = VAL(IR$)
 380 ON IR GOSUB 1000,1100,1200
 385 CLS
 390 ON BC GOSUB 2000,2100,2200
 400 GOSUB 3000
 410 GOTO 300
 500 END
1000 'SET COORDINATES
1010 INPUT "Set X1: ";X1
1020 INPUT "Set Y1: ";Y1
1030 INPUT "Set X2: ";X2
1040 INPUT "Set Y2: ";Y2
1050 RETURN
1100 'SET SWITCH
1110 INPUT SW
1200 'SET B
1210 CLS
1220 PRINT "1....No B"
1230 PRINT "2....Use B"
1240 PRINT "3....Use BF"
1250 GOSUB 3000
1260 BC = VAL(IR$)
1270 IF BC > 3 THEN GOTO 1200
1280 RETURN
```

```
2000 LINE(X1,Y1)-(X2,Y2),SW
2010 RETURN
2100 LINE(X1,Y1)-(X2,Y2),SW,B
2110 RETURN
2200 LINE(X1,Y1)-(X2,Y2),SW,BF
2210 RETURN
3000 'INKEY ROUTINE
3010 IR$=INKEY$: IF IR$="" THEN 3010
3020 RETURN
  10 CLS
  20 DEFINT A-Z
  30 INPUT "X1";X1
  40 INPUT "Y1";Y1
  50 INPUT "X2";X2
  60 INPUT "Y2";Y2
  70 INPUT "SW";SW
  80 CLS: PRINT "1...NO B": PRINT "2...B":PRINT "3...BF"
  90 IR$=INKEY$: IF IR$="" THEN 90
 100 IR = VAL(IR$):IR = INT(IR): IF IR < 0 OR IR > 3 THEN 80
 110 CLS: ON IR GOSUB 1000,2000,3000
 120 IR$=INKEY$: IF IR$="" THEN 120
 130 PRINT "CHANGE COORDINATES (1), SW (2), OR B (3)?"
 140 INPUT IR: ON IR GOTO 10, 70, 80
1000 LINE(X1,Y1)-(X2,Y2),SW
1010 RETURN
2000 LINE(X1,Y1)-(X2,Y2),SW,B
2010 RETURN
3000 LINE(X1,Y1)-(X2,Y2),SW,BF
3010 RETURN
```

Fig. 19-3. Sound program.

```
100 CH$=INKEY$:IF CH$ = "" THEN 100
200 IF CH$="Z" THEN SOUND 1174,25
210 IF CH$="X" THEN SOUND 1046,25
220 IF CH$="C" THEN SOUND 932,25
230 IF CH$="V" THEN SOUND 879,25
240 IF CH$="B" THEN SOUND 783,25
250 IF CH$="N" THEN SOUND 698,25
260 IF CH$="M" THEN SOUND 622,25
270 IF CH$="," THEN SOUND 587,25
280 GOTO 100
```

Fig. 19-4. Alarm program.

```
  10 DEFINT A-Z
  20 TIME$ ON
  30 KEY ON
  40 AL$ = "00:00:00"
 100 ON KEY GOSUB 1000,2000,3000
 200 CLS
 210 PRINT @ 50, "*** Alarm Clock ***";
 220 PRINT @ 280, "Press  F1  to set Alarm.";
 230 PRINT @ 215, TIME$;
 240 GOTO 230
1000 ' SET ALARM
1005 TIME$ OFF
1010 CLS
1020 PRINT : PRINT "Type alarm time (24 hour clock)"
1030 PRINT "(HH:MM:SS) ";
1040 LINE INPUT AL$
1050 TIME$ ON
1060 ON TIME$=AL$ GOSUB 4000
1070 GOTO 200
2000 'SNOOZE ALARM
2005 TIME$ OFF
2010 HH$ = LEFT$(AL$,2): HH = VAL(HH$)
2020 MM$ = MID$(AL$,4,2): MM = VAL(MM$)
2030 SS$ = RIGHT$(AL$,2)
2100 IF MM+10 < 60 THEN MM=MM+10: GOTO 2200
2110 MM = (MM+10)-60
2120 HH = HH+1: IF HH > 23 THEN HH = 0
2200 HH$ = STR$(HH): IF LEN(HH$) < 2 THEN HH$ = "0"+HH$
2210 MM$ = STR$(MM): IF LEN(MM$) < 2 THEN MM$ = "0"+MM$
2230 IF LEN(HH$) > 2 THEN HH$ = RIGHT$(HH$,2)
2240 IF LEN(MM$) > 2 THEN MM$ = RIGHT$(MM$,2)
2250 AL$=HH$+":"+MM$+":"+SS$
2260 TIME$ ON
2270 ON TIME$=AL$ GOSUB 4000
2280 GOTO 200
3000 GOTO 200
4000 'ALARM
4010 CLS
4015 PRINT @ 175, TIME$;
4020 PRINT @ 250, "Press  F2  for snooze.";
4030 PRINT @ 290, "Press  F3  for off.";
4040 FOR I = 1 TO 10:BEEP:NEXT
4050 CLS
4060 FOR I = 1 TO 300:NEXT
4070 GOTO 4000
```

Chapter 20

Enhancement Programs

The Model 100's BASIC is truly interactive. You can write BASIC programs to use with TEXT, TELCOM, SCHEDL, and ADDRSS. For example, you can write programs to revise and print TEXT documents, programs to sort and rearrange the data bases you use with ADDRSS and SCHEDL, and programs to communicate over the modem or RS-232.

The key to most of these interactive uses is the open command.

In this chapter you will find three programs:

- ☐ Global Search and Replace replaces every occurrence of a word or phrase in your documents.
- ☐ Address Writer can be used to update your ADRS.DO file.
- ☐ Document Uploader uses the RS-232 to upload documents from the Model 100 to a Model III. You can easily adapt it to upload to other computers.

SEARCH AND REPLACE PROGRAM

One of the most useful word processing features is search and replace. For example, if you are writing a novel and want to find every occurrence of "large mackeral" and replace it with "great white whale," you use search and replace.

The BASIC program in Fig. 20-1 opens a document, determines its length in strings, loads the documents, separates the search string, then replaces the search string with the replace string. Notice how the program uses the open command to open the document for input. It opens the document again for output. Notice too how it uses print# and input# statements to output and input data from the open files.

1. Type the program in Fig. 20-1.
2. Save the program as GSR (for Global Search and Replace).
3. Run the program and edit out any bugs.

ADDRESS WRITER

In order to take greatest advantage of the ADDRSS data base (ADRS.DO), you will want to sort it from time to time by listing all the records in alphabetical order, by ID, etc. In order to sort most effectively, it helps if you have typed each entry in an identical format. The simple BASIC program in Fig. 20-2 writes your addresses to ADRS.DO in a consistent way.

1. Type the program in Fig. 20-2.
2. Save the program as ADRSWT (for Address Write).
3. Run the program and edit out any bugs.

You can also rewrite the program for use with NOTE.DO.

Fig. 20-1. Search and Replace program (continued on page 142).

```
100 CLS:CLEAR 2000
110 DEFINT A-Z
120 MAXFILES = 1
200 'INPUT FILENAME
210 PRINT TAB(8) "*** SEARCH AND REPLACE ***"
220 FILES
230 LINE INPUT "Document? ";DD$
240 SD = INSTR(DD$,".")
250 IF SD=0 THEN DD$ = DD$+".DO"
300 'INPUT SEARCH AND REPLACE STRINGS
310 CLS
320 PRINT "Type the SEARCH string: ": GOSUB 390
330 LINE INPUT SS$:LS = LEN(SS$)
340 PRINT : PRINT "Type the REPLACE string: ": GOSUB 390
350 LINE INPUT RS$: LR = LEN(RS$)
360 IF LR = 0 THEN DS$ = "Deleting " ELSE DS$ = "Replacing "
370 IF LR = 0 THEN FP$ = "." ELSE FP$ = " with "+CHR$(34)+RS$+"."
+CHR$(34)
380 CLS: PRINT "Okay. I'm ";DS$:PRINT CHR$(34);SS$;CHR$(34);FP$:
GOTO 400
390 PRINT STRING$(40,95);: PRINT STRING$(40,8);:RETURN
400 'FIND TOTAL NUMBER OF STRINGS IN DOCUMENT
410 OPEN DD$ FOR INPUT AS 1
420 TC=0:NL=0
430 NL = NL+1
440 IF EOF(1) THEN GOTO 530
450 LINE INPUT#1, LR$
460 SP = INSTR(LR$,SS$)
470 IF SP = 0 THEN GOTO 430
```

```
480 KT=1
490 KS = SP + 1:KT =KT+1
500 SP = INSTR(KS,LR$,SS$)
510 IF SP <> 0 THEN GOTO 490
520 KT=KT+1:TC=TC+(KT*2)+1: GOTO 430
530 CLOSE
540 TT = TC+NL+1: DIM LN$(TT)
600 OPEN DD$ FOR INPUT AS 1
610 KT=1
620 IF EOF(1) THEN 710
630 LINE INPUT#1, LR$
640 SP = INSTR(LR$,SS$)
650 IF SP = 0 THEN LN$(KT) = LR$: KT=KT+1: GOTO 620
660 LL = LEN(LR$)
670 LN$(KT) = LEFT$(LR$,SP-1)
680 KT=KT+1: LN$(KT) = MID$(LR$,SP,LS)
690 KT = KT+1: LR$ = RIGHT$(LR$,(LL-SP)-LS+1): GOTO 640
710 CLOSE
720 KT=KT+1
800 'REPLACE
810 FOR I = 1 TO KT
820 IF LN$(I) = SS$ THEN LN$(I) = RS$
830 NEXT
900 KILL DD$
910 OPEN DD$ FOR OUTPUT AS 1
920 FOR I = 1 TO KT
930 PRINT#1,LN$(I);
940 NEXT
950 PRINT#1, CHR$(26)
960 CLOSE
970 PRINT: PRINT "SEARCH AND REPLACE IS COMPLETE."
980 PRINT "Do you want to seach and replace another document?"
985 PRINT "(Y/N)";
990 IR$ = INKEY$: IF IR$="" THEN 990
995 IF IR$ = "Y" THEN GOTO 100
1000 MENU
```

Fig. 20-2. Address Writer program (continued on page 143).

```
10 CLEAR 5000
20 DEFINT A-Z
30 DIM RD$(10), RR$(10)
40 MAXFILES = 1
```

```
50 FOR I = 1 TO 10
60 READ RD$(I)
70 NEXT
100 'PROGRAM DRIVER
110 CLS: PRINT TAB(12) "*** ADRS.DO UPDATE ***"
120 PRINT: PRINT "Type A to add a new address"
125 PRINT "R to return to Menu."
130 GOSUB 5000
140 IF IR$ = "R" OR IR$ = "r" THEN MENU
145 CLS
150 'INPUT A NEW RECORD
160 FOR IU = 1 TO 10
170 GOSUB 1000
175 CLS
180 NEXT IU
200 ' PRINT AND EDIT
210 I1=1: I2 = 5
220 CLS: PRINT "Here are your data items:"
230 FOR IU = I1 TO I2
240 PRINT IU;".  ";RD$(IU);": ";RR$(IU)
250 NEXT IU
260 PRINT "Do you want to change any of these (Y/N)? ";
270 GOSUB 5000
280 IF IR$ = "Y" OR IR$ = "y" THEN GOSUB 2000:GOTO 230
290 IF I1 = 1 THEN I1=6: I2=10: GOTO 220
300 GOSUB 3000
310 GOTO 110
1000 'INPUT SUBROUTINE
1010 PRINT RD$(IU);"?"
1020 LINE INPUT RR$(IU)
1030 RETURN
2000 ' EDIT SUBROUTINE
2010 PRINT @ 280, "Which one ";I1;" - ";I2;
2020 INPUT ER
2030 IF ER < I1 OR ER > I2 THEN 2020
2040 CLS: PRINT ER;". ";RR$(ER)
2050 PRINT:PRINT "Retype the ";
2060 IU = ER: GOSUB 1000
2070 CLS:RETURN
3000 'WRITE FINISHED RECORD
3010 D$ = ", "
3020 FOR I = 1 TO 10
3030 RC$=RC$+RR$(I)
3040 IF I = 4 THEN RC$ = RC$ + "."
3050 IF I = 9 THEN RC$ = RC$ + ":"
```

```
3060 IF I <> 8 AND I < 10 THEN RC$ = RC$ + D$
3070 IF I = 7 THEN RC$ = RC$ + ":"
3080 NEXT I
4000 OPEN "ADRS.DO" FOR APPEND AS 1
4010 PRINT #1, RC$
4020 CLOSE 1: RETURN
5000 ' INKEY SUBROUTINE
5010 IR$ = INKEY$: IF IR$ = "" THEN 5010
5020 RETURN
6000 ' DATA
6010 DATA "Code", "Last name", "First name", "Middle initial", "S
treet", "City", "State (2 letters)", "Area code ", "Phone number"
, "Description"
```

Appendix

ASCII Code

Decimal	Printed Character	Keyboard Character	Decimal	Printed Character	Keyboard Character
0		CTRL @	17		CTRL Q
1		CTRL A	18		CTRL R
2		CTRL B	19		CTRL S
3		CTRL C	20		CTRL T
4		CTRL D	21		CTRL U
5		CTRL E	22		CTRL V
6		CTRL F	23		CTRL W
7		CTRL G	24		CTRL X
8		CTRL H	25		CTRL Y
9		CTRL I	26		CTRL Z
10		CTRL J	27		ESC
11		CTRL K	28		→
12		CTRL L	29		←
13		CTRL M	30		↑
14		CTRL N	31		↓
15		CTRL O	32		SPACEBAR
16		CTRL P	33	!	!

Decimal	Printed Character	Keyboard Character
34	”	”
35	#	#
36	$	$
37	%	%
38	&	&
39	’	’
40	(	(
41	)	)
42	*	*
43	+	+
44	,	,
45	-	-
46	.	.
47	/	/
48	0	0
49	1	1
50	2	2
51	3	3
52	4	4
53	5	5
54	6	6
55	7	7
56	8	8
57	9	9
58	:	:
59	;	;
60	<	<
61	=	=
62	>	>
63	?	?
64	@	@
65	A	A
66	B	B
67	C	C
68	D	D
69	E	E
70	F	F
71	G	G
72	H	H
73	I	I
74	J	J
75	K	K
76	L	L
77	M	M
78	N	N
79	O	O
80	P	P
81	Q	Q
82	R	R
83	S	S
84	T	T
85	U	U
86	V	V
87	W	W
88	X	X
89	Y	Y
90	Z	Z
91	[	[
92	/	GRPH -
93	]	]
94	•	•
95	—	—
96	\	GRAPH [
97	a	A
98	b	B
99	c	C
100	d	D
101	e	E
102	f	F
103	g	G
104	h	H
105	i	I
106	j	J
107	k	K
108	l	L
109	m	M

* For uppercase letters A-Z, press SHIFT or CAPS LOCK before pressing the Keyboard Character.

Decimal	Printed Character	Keyboard Character	Decimal	Printed Character	Keyboard Character
110	n	N	148	🯅	GRPH w
111	o	O	149	♂	GRPH b
112	p	P	150	♀	GRPH n
113	q	Q	151	%	GRPH .
114	r	R	152	↑	GRPH o
115	s	S	153	↓	GRPH ,
116	t	T	154	→	GRPH l
117	u	U	155	←	GRPH k
118	v	V	156	♧	GRPH 2
119	w	W	157	♢	GRPH 3
120	x	X	158	♡	GRPH 4
121	y	Y	159	♤	GRPH 5
122	z	Z	160	ʼ	CODE '
123	{	GRPH 9	161	à	CODE x
124	|	GRPH —	162	ç	CODE c
125	}	GRPH 0	163	£	GRPH 8
126		GRPH]	164	‵	CODE "
127		DEL	165	μ	CODE M
128	☎	GRPH p	166	°	CODE)
129	⌂	GRPH m	167	▼	CODE —
130	(×	GRPH f	168	†	CODE +
131	⌐	GRPH x	169	∽	CODE s
132	⌘	GRPH c	170	🅿	CODE R
133	✈	GRPH a	171	🅲	CODE C
134	⌂	GRPH h	172	¼	CODE p
135	▣	GRPH t	173	¾	CODE ;
136	i	GRPH I	174	½	CODE /
137	√	GRPH r	175	¶	CODE 0
138	≠	GRPH /	176	¥	GRPH 7
139	Σ	GRPH s	177	A	CODE A
140	≈	GRPH '	178	Ö	CODE O
141	±	GRPH =	179	Ü	CODE U
142	∫	GRPH i	180	¢	GRPH 6
143	◀	GRPH e	181	-	CODE [
144	⩎	GRPH y	182	ä	CODE a
145	∩	GRPH u	183	ŏ	CODE o
146	↕	GRPH ;	184	ü	CODE u
147	🯅	GRPH q	185	B	CODE S

* For lowercase letters a-z, be sure CAPS LOCK is not pressed "down."

Decimal	**Printed Character**	**Keyboard Character**	**Decimal**	**Printed Character**	**Keyboard Character**
186	™	CODE T	221	U ′	CODE <
187	é	CODE d	222	E ′	CODE V
188	ù	CODE ,	223	A ′	CODE X
189	è	CODE v	224		GRPH Z
190	¨	CODE =	225	▘ (upper left)	GRPH !
191	£	CODE F	226	▝ (upper right)	GRPH@
192	â	CODE I	227	▖ (lower left)	GRPH #
193	ê	CODE 3	228	▗ (lower right)	GRPH $
194	î	CODE 8	229	▚	GRPH %
195	ô	CODE 9	230	▞	GRPH '
196	û	CODE 7	231	▔ (upper)	GRPH Q
197		CODE -	232	▁ (lower)	GRPH W
198	ë	CODE e	233	▌ (left)	GRPH E
199	ï	CODE i	234	▐ (right)	GRPH R
200	á	CODE q	235	▛	GRPH A
201	í	CODE k	236	▜	GRPH S
202	ó	CODE I	237	▙	GRPH D
203	ú	CODE j	238	▟	GRPH F
204	ý	CODE y	239	█	GRPH X
205	ñ	CODE n	240	┌	GRPH U
206	ã	CODE z	241	─	GRPH P
207	õ	CODE .	242	┐	GRPH O
208	Â	CODE !	243	┬	GRPH I
209	Ê	CODE #	244	├	GRPH J
210	Î	CODE *	245	│	GRPH :
211	Ô	CODE (	246	└	GRPH M
212	Û	CODE &	247	┘	GRPH >
213	Ï	CODE I	248	┴	GRPH <
214	Ë	CODE E	249	┤	GRPH L
215	É	CODE D	250	+	GRPH K
216	Á	CODE Q	251	◤	GRPH H
217	Í	CODE K	252	◢	GRPH T
218	Ó	CODE L	253	◥	GRPH G
219	Ú	CODE J	254	◣	GRPH Y
220	Ý	CODE Y	255	▒	GRPH C

Index

A

Ac adaptor, 3, 17, 19
Acoustic coupler, 19, 76
Address book, 60
Address file, 26
Address of highest byte, 113
Address writer program, 141, 142
ADDRSS, 26, 57
ADDRSS and SCHEDL, differences between, 60
ADRS.DO document, 62-63
Alarm clock program, 135, 139
ANS/ORIG switch, 19, 75
Appointment file, 27
Appointments, 60
Arrow, left-pointing display, 35
Arrow keys, 16-17, 39
ASCII, 32, 72, 115, 145
Auto command, 107
Auto dial feature, 28

B

.BA, 3
.BA suffix, 101
Bar code wand connector, 18
BASIC, 21, 99
BASIC, invoking, 101
BASIC, learning, ix, 99
BASIC, Model III, 91
BASIC interactions, 23
BASIC programs, working with, 114
Battery requirements, 3
Baud, 73, 74
Beep command, 125
Binary, 72
Bits, 72
Block commands, 24
Blocks, copying and cutting, 43
Boldface, 24
Box, filling in the, 128
Boxes, drawing, 22, 127-128
BREAK/PAUSE, 16
Buffer, 43
Bulletin boards, 71
Bye, 28, 80
Byte, 72

C

Call command, 28
Caps lock key, 15, 37
Carriage return indicator, 36
Carriage returns, 35
Cassette, saving programs on, 115
Cassette interface, 18
Cassette recorder, 8
Character positions, 126
Clear command, 111, 112
Clock, real time, 5, 105
Close command, 130
CMOS, 17
CODE, 15
Code key, 15, 38
Colons, use of, 26, 63
Comfile program, 92
Commands, direct, 102
Commands to work with function keys, 108
Communication, 71
Communication, computer-to-computer, 73, 89
Communication, terminating, 80
Communications parameters, 73
Communications parameters, setting, 28, 77
Comprint program, 92, 95
CompuServe, 81
CompuServe, logging on to, 82
CompuServe, setting parameters for communicating with, 77
CompuServe, using, 85
CompuServe services, finding, 87
CompuServe Start-Up Kit, 82
Computers, connecting, 73
Computer-to-computer communication, 73, 89
Comread program, 92, 95
Connections, 17

Cont command, 107
Control key, 15
Control key with cursor movement keys, 41
Copy feature, 24
Copying a document, 45, 50
CPU, 5
CTRL, 15
Cursor movement commands, 24, 39
Cursor movement keys, 16
Cut feature, 24

D

Data, sending and receiving, 74
Data base, 59, 81
Data base, creating a, 62
Data base, sorting a, 65
Data base, updating a, 63, 65
Data base management, 57, 59
Data base managers, 26
Data base planning, 61
Data bases, commercial, 71
Date, setting the, 105, 106
Date$, 105
Dc 6V connection slot, 17
Deadlines, keeping track of, 60
Default RS-232 settings for Model III, 90
DEL BKSP, 15
Delete/backspace key, 15, 24, 38
Deleting a document, 45
Dialer, automatic, 27
DIR/ACP switch, 18, 76
Disk drive, portable, 7, 8
DISP, 12
Display, 3, 5, 12
Display, printing the screen, 44
Display contrast, 12
Display control, 126
Display (DISP) control wheel, 12
Display viewing angle, 12
Display window size, 12
.DO, 3
Document, 3
Document, deleting a, 45, 46
Document, exiting a, 38
Document, renaming a, 45
Document, typing a, 33
Document, typing the, 50
Documentation, 5
Document names, 34
Documents, saving, 32
Dot matrix, 12
Dow Jones, 71
DOWN, 28
Download, 28, 74, 79, 87
Duplex, 28, 74, 79

E

Echo, 28, 79
Edit command, 23, 118
Editing, 118
Editing, text, ix
Editing files, 24
Editing programs, 23
Enhancements, program, 140
Enter key, 37
Entry mode, 76
Entry mode, TELCOM, 28
Epson, viii, 7
Errors, typographical, 24
ESC, 15
Escape key, 15
Expense account file, 27
Expenses, 60

F

File, 103
File I/O, 130
File management, 55
File number, 132
Files, 3, 22, 32
Files, kinds of, 130
Files command, 110
Find command, 25, 28
Find key, 43
Foreign-language alphabets, 12
FOR . . . NEXT loop, 22
FRE command, 111, 112
Full, 28
Function key definitions, listing, 108
Function key definitions, restoring, 109
Function keys, commands to work with, 108
Function keys, defining the, 99, 108, 109
Function keys, displaying the, 103
Function keys in BASIC, 103

G

Gavilan, viii
Getting Started with TRS-80 BASIC, ix
Global search, ix
Graphics, 12
Graphics key, 15, 38
GRPH, 15

H

HIMEM command, 111, 113
Housekeeping, 105
Housekeeping functions, 99
Housekeeping program, 134, 135

I

Information utilities, 82
Input#, 130, 140
Insert feature, 24
Inserting characters, 41
Interface ports, 5
Interpreter, BASIC, 99
Interrupt commands, 21, 121
Interrupts, 121
IPL command, 106

K

Keyboard, 3, 5, 15
Key command, 108
Key list command, 103
Kill command, 48, 110

L

LABEL, 16
Label key, 39
LCD, viii, 12
Line command, 126
Line demo program, 134, 137
Lines, drawing, 22, 126
Line status, 74
List, 104, 117
Listing a resident program, 117
Load, 103, 116
Loading programs, 116
Log on, 73
Log-on procedure, 83
Low battery indicator, 3, 19

M

Machine control, 99
Margins, setting, 24
Maxfiles command, 130, 131
Maxram, 111
Memory, 5, 17
Memory, free, 111
Memory, managing, 105, 111
Memory switch, 19
Menu, 104, 108
Menu, cleaning up the, 105
Menu command, 106
Menu/directory, 3
Menu program, 92, 96
Microprocessor, 17
Microprocessor, 8085, viii
Microsoft, 21, 99
Model 100 and CompuServe, 82
Model 100 built-in programs, enhancing the, 99

Model III and Model 100, communication between the, 90
Model III communication, setting parameters for, 90
Model III programs, 92-97
Model III, setting parameters for communicating with, 77
Model II and Model 100, communication between the, 89
Modem, 73, 81
Modem, saving programs over the, 115
Modem, using the, 75-76
Modem communication, 71, 73
Modem jack, direct connect, 18
Modulator/demodulator, 73

N

Name command, 110, 111
NOTE.DO document, 62
Null modem adaptor, 18, 76, 90
NUM, 15
Number key, 15, 38
Numeric expression, 112
Numeric keypad, 15

O

ON COM, 121, 125
ON COM command, 21
On Error command, 21, 121
ON KEY, 121, 124
ON KEY command, 21
ON MDM, 121, 124
ON MDM command, 21
On/off switch, 18
ON TIME$, 121, 123
ON TIME$ command, 21
Open command, 22, 128, 130, 131
Opening a file, 33
Options, Model 100, 5
Originate/answer, 75
Osborne portable, viii, 7

P

Parity, 74
PASTE, 16
Paste feature, 24
Pasting new copy, 52
Peek, 111
Pitch, varying, 22
Pixel positions, screen, 127
Pixels, 22
Poke, 111
Power command, 21, 106
Power commands, 105, 106
Power CONT command, 107
Power off, controlling automatic, 107
Power off command, 107, 108
Power supply, 5, 17
Present command, 126
PREV, 28
Previous, 28
Price, Model 100, 5, 7
PRINT, 16
Print @ command, 126
Print command, 126
Print command key, 25
Printer interface, 44
Printer interface, parallel, 18
Printer, portable, 7
Printer, saving programs on the, 116
Printing a document, 55
Printing width default, 44
Print#, 130, 140
Print tab command, 126
Program, BASIC, 3
Program, writing a, 114
Program enhancements, 99, 140
Programmable function keys, 15
Program names, 115
Programs, built-in, viii
Programs, demonstration, 134
Programs, interactive, 140
Programs, saving, 32, 114
Programs, working with BASIC, 116
Programs as documents, saving, 115
Programs into document, turning, 120
PSET command, 126

Q

QWERTY keyboard, 3, 15

R

RAM, 3, 17
RAM files, 3, 32
RAM files, managing, 105, 109
Record, finding a, 65
Records, list of, 59
Records, pointers on typing, 62, 63
Reformatting, ix
Renaming a document, 48
Reset switch, 18
Resume command, 22, 106, 107 108
Reverse field, 12
Reverse video, 12, 24
Revising a document, 52
ROM, 3, 17
RS-232C, saving programs through, 116
RS-232C, sending characters over the, 91
RS-232C interface, 17, 20, 73, 89
Run, 104
Running a program from the menu, 101
Running programs, 118
Run options, 118

S

Save, 104
Saving programs, 115
Search and replace, 24
Search and replace program, 140, 141
Searching records, 59
SCHEDL, 26, 27, 57
SCHEDL and ADDRESS, differences between, 60
Schedules, 60
Screen, displaying programs on the, 116
Scrolling, 35-36
Sharp, viii
Shift key, 37
Shift key with cursor movement keys, 40
Size, Model 100, 5
Software, 5
Sorting, 59, 141
Sorting files, 24
Sound command, 12, 22
Sound generator, 5
Sound off command, 125
Sound on command, 125
Sound program, 134, 138
Source, The, 71
Special-purpose keys, 15
Specifications, Model 100, 5
Start bit, 72
Start-up, 3
Start-up, automatic, 106
Start-up commands, 105, 106
Start-Up Kit, CompuServe, 82
STAT, 28
Status command, 28
Stop bit, 72, 74
String, 43
String expression, 112
String storage, setting memory aside for, 113
Switch commands, 127
Switches, Model 100, 18

T

Tab key, 37
Tabs, setting, 24, 37
Tandy, viii
TELCOM, 27

TELCOM and ADRS.DO, 63
TELCOM modes, 28, 76
Telecommunications, 69
Telecommunications, disconnecting, 28
Telenet, 81, 87
Telephone dialing, 77-78
Telephone numbers, dialing, 63, 76
Term command, 28
Terminal mode, 76, 77, 78
Terminal mode, TELCOM, 28
Term program, 92, 94
TEXT, 24
Text, printing, 25, 44
Text, revising, 41
TEXT, selecting, 32
Text blocks, copying and cutting, 43
Text blocks, revising, 43
TEXT with cassette, 48
Time, setting the, 105, 106
Time$, 105
Time$ with power off, 108
Tone duration, varying, 22
TRSDOS SETCOM, 90
Tymnet, 81, 87
Typing a document, 50
Typing hints, 36

U

Underscores, 24
UP, 28
Upload, 28, 73, 74, 79
UPM1 program, 92, 93
Uppercase characters, 24
Users, Model 100, 9

V

VARPTR, 111
VIDTEX, 91

W

Wand, Hewlett-Packard, 18
Warranty, Model 100, 5
Weight, Model 100, 5
Word length, 74
Word processing, 12, 24, 44
Wraparound display feature, 35
Writing a program, 120

TRS-80™ Model 100 — A User's Guide

by Joseph Coleman

Beginning where standard operator's manuals leave off, this timely and meticulously thorough handbook gives you a guided tour through all of the amazingly sophisticated hardware and software features of this new generation portable—a machine the size of an ordinary loose leaf notebook!

It shows you how to take maximum advantage of such features as: the eight-line liquid crystal display, memory retention that functions even when the machine is turned off, built-in interfaces for cassette, bar-code reading devices, direct connect telephone modem, and more.

Each of the built-in software programs is covered in detail (TEXT, SCHEDULE, ADDRESS, and TELCOM) with expert advice on how to use and adapt them to your own particular application needs. You'll also get a hands-on introduction to Model 100 BASIC . . . learn how to use this BASIC for Model 100 "housekeeping" . . . get a look at programming techniques and "tricks" that let you get more powerful problem solving action from your machine . . . and find out about the unique features of the Model 100's enhanced version of MICROSOFT BASIC.

Plus, you'll find several demonstration programs designed to help you get the "feel" of your computer's programming versatility . . . even get a BASIC program you can use to perform a Global Search and Replace on documents used with the built-in TEXT editor program.

If you own a TRS-80 Model 100—or are trying to decide whether or not the 100 offers what you need in a microcomputer—this excellent sourcebook is a must-read selection!

Joseph Coleman has written and designed many educational and technical publications for Tandy/Radio Shack, Phillips/Micom, and other microcomputer companies.

Cover photograph courtesy of Radio Shack, A division of Tandy Corporation.

OTHER POPULAR TAB BOOKS OF INTEREST

Troubleshooting and Repairing Personal Computers (No. 1539—$13.95 paper; $19.95 hard)

Professional Programming Techniques—Starting with the BASICs (No. 1428—$10.95 paper; $17.95 hard)

TAB TAB BOOKS Inc.
Blue Ridge Summit, Pa. 17214

Send for FREE TAB Catalog describing over 750 current titles in print.

FPT > $15.50

ISBN 0-8306-1651-9

PRICES HIGHER IN CANADA

1495-1283